

NVIDIA Performance Primitives (NPP)

Version 4.1

January 10, 2012

Contents

1	NVIDIA Performance Primitives	1
1.1	What is NPP?	1
1.2	Documentation	1
1.3	Technical Specifications	1
1.4	Files	2
1.4.1	Header Files	2
1.4.2	Library Files	2
1.5	Supported NVIDIA Hardware	2
2	General API Conventions	3
2.1	Memory Management	4
2.1.1	Scratch Buffer and Host Pointer	4
2.2	Function Naming	4
2.3	Integer Result Scaling	5
3	Signal-Processing Specific API Conventions	7
3.1	Signal Data	8
3.1.1	Parameter Names for Signal Data	8
3.1.1.1	Source Signal Pointer	8
3.1.1.2	Destination Signal Pointer	8
3.1.1.3	In-Place Signal Pointer	8
3.1.2	Signal Data Alignment Requirements	9
3.1.3	Signal Data Related Error Codes	9
3.2	Signal Length	9
3.2.1	Length Related Error Codes	9
4	Imaging-Processing Specific API Conventions	11
4.1	Function Naming	12
4.2	Image Data	12

4.2.1	Line Step	13
4.2.2	Parameter Names for Image Data	13
4.2.2.1	Passing Source-Image Data	13
4.2.2.2	Passing Destination-Image Data	14
4.2.2.3	Passing In-Place Image Data	14
4.2.2.4	Passing Mask-Image Data	14
4.2.3	Image Data Alignment Requirements	15
4.2.4	Image Data Related Error Codes	15
4.3	Region-of-Interest (ROI)	15
4.3.1	ROI Related Error Codes	16
4.4	Masked Operation	16
4.5	Channel-of-Interest API	16
4.5.1	Select-Channel Source-Image Pointer	16
4.5.2	Select-Channel Destination-Image Pointer	17
4.6	Geometric Transform API Specifics	17
4.6.1	Geometric Transforms and ROIs	17
4.6.2	Pixel Interpolation	17
4.6.3	Rotate specific Error Codes	18
5	Module Index	19
5.1	Modules	19
6	Data Structure Index	21
6.1	Data Structures	21
7	Module Documentation	23
7.1	NPP Core	23
7.1.1	Detailed Description	23
7.1.2	Function Documentation	24
7.1.2.1	nppGetGpuComputeCapability	24
7.1.2.2	nppGetGpuName	24
7.1.2.3	nppGetGpuNumSMs	24
7.1.2.4	nppGetLibVersion	24
7.1.2.5	nppGetMaxThreadsPerBlock	24
7.1.2.6	nppGetMaxThreadsPerSM	25
7.1.2.7	nppGetStream	25
7.1.2.8	nppSetStream	25
7.2	NPP Type Definitions and Constants	26

7.2.1	Define Documentation	29
7.2.1.1	NPP_MAX_16S	29
7.2.1.2	NPP_MAX_16U	29
7.2.1.3	NPP_MAX_32S	30
7.2.1.4	NPP_MAX_32U	30
7.2.1.5	NPP_MAX_64S	30
7.2.1.6	NPP_MAX_64U	30
7.2.1.7	NPP_MAX_8S	30
7.2.1.8	NPP_MAX_8U	30
7.2.1.9	NPP_MAXABS_32F	30
7.2.1.10	NPP_MAXABS_64F	30
7.2.1.11	NPP_MIN_16S	30
7.2.1.12	NPP_MIN_16U	30
7.2.1.13	NPP_MIN_32S	30
7.2.1.14	NPP_MIN_32U	31
7.2.1.15	NPP_MIN_64S	31
7.2.1.16	NPP_MIN_64U	31
7.2.1.17	NPP_MIN_8S	31
7.2.1.18	NPP_MIN_8U	31
7.2.1.19	NPP_MINABS_32F	31
7.2.1.20	NPP_MINABS_64F	31
7.2.2	Enumeration Type Documentation	31
7.2.2.1	NppCmpOp	31
7.2.2.2	NppGpuComputeCapability	31
7.2.2.3	NppHintAlgorithm	32
7.2.2.4	NppiAxis	32
7.2.2.5	NppiBorderType	32
7.2.2.6	NppiInterpolationMode	32
7.2.2.7	NppRoundMode	33
7.2.2.8	NppStatus	33
7.3	Basic NPP Data Types	35
7.3.1	Typedef Documentation	36
7.3.1.1	Npp16s	36
7.3.1.2	Npp16u	36
7.3.1.3	Npp32f	36
7.3.1.4	Npp32s	36

7.3.1.5	Npp32u	36
7.3.1.6	Npp64f	36
7.3.1.7	Npp64s	36
7.3.1.8	Npp64u	36
7.3.1.9	Npp8s	36
7.3.1.10	Npp8u	36
7.4	NPP Image Processing	37
7.4.1	Enumeration Type Documentation	38
7.4.1.1	NppiAlphaOp	38
7.5	Memory Management	39
7.5.1	Detailed Description	41
7.5.2	Function Documentation	41
7.5.2.1	nppiFree	41
7.5.2.2	nppiMalloc_16s_C1	41
7.5.2.3	nppiMalloc_16s_C2	42
7.5.2.4	nppiMalloc_16s_C4	42
7.5.2.5	nppiMalloc_16sc_C1	42
7.5.2.6	nppiMalloc_16sc_C2	42
7.5.2.7	nppiMalloc_16sc_C3	43
7.5.2.8	nppiMalloc_16sc_C4	43
7.5.2.9	nppiMalloc_16u_C1	43
7.5.2.10	nppiMalloc_16u_C2	44
7.5.2.11	nppiMalloc_16u_C3	44
7.5.2.12	nppiMalloc_16u_C4	44
7.5.2.13	nppiMalloc_32f_C1	44
7.5.2.14	nppiMalloc_32f_C2	45
7.5.2.15	nppiMalloc_32f_C3	45
7.5.2.16	nppiMalloc_32f_C4	45
7.5.2.17	nppiMalloc_32fc_C1	46
7.5.2.18	nppiMalloc_32fc_C2	46
7.5.2.19	nppiMalloc_32fc_C3	46
7.5.2.20	nppiMalloc_32fc_C4	46
7.5.2.21	nppiMalloc_32s_C1	47
7.5.2.22	nppiMalloc_32s_C3	47
7.5.2.23	nppiMalloc_32s_C4	47
7.5.2.24	nppiMalloc_32sc_C1	48

7.5.2.25	nppiMalloc_32sc_C2	48
7.5.2.26	nppiMalloc_32sc_C3	48
7.5.2.27	nppiMalloc_32sc_C4	48
7.5.2.28	nppiMalloc_8u_C1	49
7.5.2.29	nppiMalloc_8u_C2	49
7.5.2.30	nppiMalloc_8u_C3	49
7.5.2.31	nppiMalloc_8u_C4	50
7.6	Data-Exchange and Initialization	51
7.6.1	Detailed Description	66
7.6.2	Function Documentation	67
7.6.2.1	nppiConvert_16s32f_C1R	67
7.6.2.2	nppiConvert_16s32s_C1R	67
7.6.2.3	nppiConvert_16s8u_AC4R	67
7.6.2.4	nppiConvert_16s8u_C1R	68
7.6.2.5	nppiConvert_16s8u_C4R	68
7.6.2.6	nppiConvert_16u32f_C1R	68
7.6.2.7	nppiConvert_16u32s_C1R	69
7.6.2.8	nppiConvert_16u8u_AC4R	69
7.6.2.9	nppiConvert_16u8u_C1R	70
7.6.2.10	nppiConvert_16u8u_C4R	70
7.6.2.11	nppiConvert_32f16s_C1R	70
7.6.2.12	nppiConvert_32f16u_C1R	71
7.6.2.13	nppiConvert_32f8u_C1R	71
7.6.2.14	nppiConvert_8u16s_AC4R	72
7.6.2.15	nppiConvert_8u16s_C1R	72
7.6.2.16	nppiConvert_8u16s_C4R	72
7.6.2.17	nppiConvert_8u16u_AC4R	73
7.6.2.18	nppiConvert_8u16u_C1R	73
7.6.2.19	nppiConvert_8u16u_C4R	73
7.6.2.20	nppiConvert_8u32f_C1R	74
7.6.2.21	nppiCopy_16s_AC4MR	74
7.6.2.22	nppiCopy_16s_AC4R	75
7.6.2.23	nppiCopy_16s_C1C3R	75
7.6.2.24	nppiCopy_16s_C1C4R	75
7.6.2.25	nppiCopy_16s_C1MR	76
7.6.2.26	nppiCopy_16s_C1R	76

7.6.2.27	nppiCopy_16s_C3C1R	76
7.6.2.28	nppiCopy_16s_C3CR	77
7.6.2.29	nppiCopy_16s_C3MR	77
7.6.2.30	nppiCopy_16s_C3P3R	77
7.6.2.31	nppiCopy_16s_C4C1R	78
7.6.2.32	nppiCopy_16s_C4CR	78
7.6.2.33	nppiCopy_16s_C4MR	78
7.6.2.34	nppiCopy_16s_C4P4R	79
7.6.2.35	nppiCopy_16s_C4R	79
7.6.2.36	nppiCopy_16s_P3C3R	79
7.6.2.37	nppiCopy_16s_P4C4R	80
7.6.2.38	nppiCopy_16sc_AC4R	80
7.6.2.39	nppiCopy_16sc_C1R	80
7.6.2.40	nppiCopy_16sc_C2R	81
7.6.2.41	nppiCopy_16sc_C3R	81
7.6.2.42	nppiCopy_16sc_C4R	81
7.6.2.43	nppiCopy_16u_AC4MR	82
7.6.2.44	nppiCopy_16u_AC4R	82
7.6.2.45	nppiCopy_16u_C1C3R	82
7.6.2.46	nppiCopy_16u_C1C4R	83
7.6.2.47	nppiCopy_16u_C1MR	83
7.6.2.48	nppiCopy_16u_C1R	83
7.6.2.49	nppiCopy_16u_C3C1R	84
7.6.2.50	nppiCopy_16u_C3CR	84
7.6.2.51	nppiCopy_16u_C3MR	84
7.6.2.52	nppiCopy_16u_C3P3R	85
7.6.2.53	nppiCopy_16u_C4C1R	85
7.6.2.54	nppiCopy_16u_C4CR	85
7.6.2.55	nppiCopy_16u_C4MR	86
7.6.2.56	nppiCopy_16u_C4P4R	86
7.6.2.57	nppiCopy_16u_C4R	86
7.6.2.58	nppiCopy_16u_P3C3R	87
7.6.2.59	nppiCopy_16u_P4C4R	87
7.6.2.60	nppiCopy_32f_AC4MR	87
7.6.2.61	nppiCopy_32f_AC4R	88
7.6.2.62	nppiCopy_32f_C1C3R	88

7.6.2.63	nppiCopy_32f_C1C4R	88
7.6.2.64	nppiCopy_32f_C1MR	89
7.6.2.65	nppiCopy_32f_C1R	89
7.6.2.66	nppiCopy_32f_C3C1R	89
7.6.2.67	nppiCopy_32f_C3CR	90
7.6.2.68	nppiCopy_32f_C3MR	90
7.6.2.69	nppiCopy_32f_C3P3R	90
7.6.2.70	nppiCopy_32f_C4C1R	91
7.6.2.71	nppiCopy_32f_C4CR	91
7.6.2.72	nppiCopy_32f_C4MR	91
7.6.2.73	nppiCopy_32f_C4P4R	92
7.6.2.74	nppiCopy_32f_C4R	92
7.6.2.75	nppiCopy_32f_P3C3R	92
7.6.2.76	nppiCopy_32f_P4C4R	93
7.6.2.77	nppiCopy_32fc_AC4R	93
7.6.2.78	nppiCopy_32fc_C1R	93
7.6.2.79	nppiCopy_32fc_C2R	94
7.6.2.80	nppiCopy_32fc_C3R	94
7.6.2.81	nppiCopy_32fc_C4R	94
7.6.2.82	nppiCopy_32s_AC4MR	95
7.6.2.83	nppiCopy_32s_AC4R	95
7.6.2.84	nppiCopy_32s_C1C3R	95
7.6.2.85	nppiCopy_32s_C1C4R	96
7.6.2.86	nppiCopy_32s_C1MR	96
7.6.2.87	nppiCopy_32s_C1R	96
7.6.2.88	nppiCopy_32s_C3C1R	97
7.6.2.89	nppiCopy_32s_C3CR	97
7.6.2.90	nppiCopy_32s_C3MR	97
7.6.2.91	nppiCopy_32s_C3P3R	98
7.6.2.92	nppiCopy_32s_C4C1R	98
7.6.2.93	nppiCopy_32s_C4CR	98
7.6.2.94	nppiCopy_32s_C4MR	99
7.6.2.95	nppiCopy_32s_C4P4R	99
7.6.2.96	nppiCopy_32s_C4R	99
7.6.2.97	nppiCopy_32s_P3C3R	100
7.6.2.98	nppiCopy_32s_P4C4R	100

7.6.2.99 nppiCopy_32sc_AC4R	100
7.6.2.100 nppiCopy_32sc_C1R	101
7.6.2.101 nppiCopy_32sc_C2R	101
7.6.2.102 nppiCopy_32sc_C3R	101
7.6.2.103 nppiCopy_32sc_C4R	102
7.6.2.104 nppiCopy_8s_AC4R	102
7.6.2.105 nppiCopy_8s_C1R	102
7.6.2.106 nppiCopy_8s_C2R	103
7.6.2.107 nppiCopy_8s_C3R	103
7.6.2.108 nppiCopy_8s_C4R	103
7.6.2.109 nppiCopy_8u_AC4MR	104
7.6.2.110 nppiCopy_8u_AC4R	104
7.6.2.111 nppiCopy_8u_C1C3R	104
7.6.2.112 nppiCopy_8u_C1C4R	105
7.6.2.113 nppiCopy_8u_C1MR	105
7.6.2.114 nppiCopy_8u_C1R	105
7.6.2.115 nppiCopy_8u_C3C1R	106
7.6.2.116 nppiCopy_8u_C3CR	106
7.6.2.117 nppiCopy_8u_C3MR	106
7.6.2.118 nppiCopy_8u_C3P3R	107
7.6.2.119 nppiCopy_8u_C4C1R	107
7.6.2.120 nppiCopy_8u_C4CR	107
7.6.2.121 nppiCopy_8u_C4MR	108
7.6.2.122 nppiCopy_8u_C4P4R	108
7.6.2.123 nppiCopy_8u_C4R	108
7.6.2.124 nppiCopy_8u_P3C3R	109
7.6.2.125 nppiCopy_8u_P4C4R	109
7.6.2.126 nppiCopyConstBorder_32s_C1R	109
7.6.2.127 nppiCopyConstBorder_8u_AC4R	110
7.6.2.128 nppiCopyConstBorder_8u_C1R	110
7.6.2.129 nppiCopyConstBorder_8u_C4R	111
7.6.2.130 nppiSet_16s_AC4MR	111
7.6.2.131 nppiSet_16s_AC4R	112
7.6.2.132 nppiSet_16s_C1MR	112
7.6.2.133 nppiSet_16s_C1R	112
7.6.2.134 nppiSet_16s_C2R	113

7.6.2.135 nppiSet_16s_C4CR	113
7.6.2.136 nppiSet_16s_C4MR	114
7.6.2.137 nppiSet_16s_C4R	114
7.6.2.138 nppiSet_16sc_AC4R	114
7.6.2.139 nppiSet_16sc_C1R	115
7.6.2.140 nppiSet_16sc_C2R	115
7.6.2.141 nppiSet_16sc_C3R	115
7.6.2.142 nppiSet_16sc_C4R	116
7.6.2.143 nppiSet_16u_AC4MR	116
7.6.2.144 nppiSet_16u_AC4R	116
7.6.2.145 nppiSet_16u_C1MR	117
7.6.2.146 nppiSet_16u_C1R	117
7.6.2.147 nppiSet_16u_C2R	117
7.6.2.148 nppiSet_16u_C4CR	118
7.6.2.149 nppiSet_16u_C4MR	118
7.6.2.150 nppiSet_16u_C4R	118
7.6.2.151 nppiSet_32f_AC4MR	119
7.6.2.152 nppiSet_32f_AC4R	119
7.6.2.153 nppiSet_32f_C1MR	120
7.6.2.154 nppiSet_32f_C1R	120
7.6.2.155 nppiSet_32f_C4CR	120
7.6.2.156 nppiSet_32f_C4MR	121
7.6.2.157 nppiSet_32f_C4R	121
7.6.2.158 nppiSet_32fc_AC4R	121
7.6.2.159 nppiSet_32fc_C1R	122
7.6.2.160 nppiSet_32fc_C2R	122
7.6.2.161 nppiSet_32fc_C3R	122
7.6.2.162 nppiSet_32fc_C4R	123
7.6.2.163 nppiSet_32s_AC4MR	123
7.6.2.164 nppiSet_32s_AC4R	123
7.6.2.165 nppiSet_32s_C1MR	124
7.6.2.166 nppiSet_32s_C1R	124
7.6.2.167 nppiSet_32s_C4CR	125
7.6.2.168 nppiSet_32s_C4MR	125
7.6.2.169 nppiSet_32s_C4R	125
7.6.2.170 nppiSet_32sc_AC4R	126

7.6.2.171	nppiSet_32sc_C1R	126
7.6.2.172	nppiSet_32sc_C2R	126
7.6.2.173	nppiSet_32sc_C3R	127
7.6.2.174	nppiSet_32sc_C4R	127
7.6.2.175	nppiSet_8s_AC4R	127
7.6.2.176	nppiSet_8s_C1R	128
7.6.2.177	nppiSet_8s_C2R	128
7.6.2.178	nppiSet_8s_C3R	128
7.6.2.179	nppiSet_8s_C4R	129
7.6.2.180	nppiSet_8u_AC4MR	129
7.6.2.181	nppiSet_8u_AC4R	129
7.6.2.182	nppiSet_8u_C1MR	130
7.6.2.183	nppiSet_8u_C1R	130
7.6.2.184	nppiSet_8u_C4CR	130
7.6.2.185	nppiSet_8u_C4MR	131
7.6.2.186	nppiSet_8u_C4R	131
7.6.2.187	nppiSwapChannels_8u_C4IR	131
7.6.2.188	nppiTranspose_8u_C1R	132
7.7	Arithmetic and Logical Operations	133
7.7.1	Function Documentation	214
7.7.1.1	nppiAbs_16s_AC4IR	214
7.7.1.2	nppiAbs_16s_AC4R	215
7.7.1.3	nppiAbs_16s_C1IR	215
7.7.1.4	nppiAbs_16s_C1R	215
7.7.1.5	nppiAbs_16s_C3IR	216
7.7.1.6	nppiAbs_16s_C3R	216
7.7.1.7	nppiAbs_16s_C4IR	216
7.7.1.8	nppiAbs_16s_C4R	217
7.7.1.9	nppiAbs_32f_AC4IR	217
7.7.1.10	nppiAbs_32f_AC4R	217
7.7.1.11	nppiAbs_32f_C1IR	218
7.7.1.12	nppiAbs_32f_C1R	218
7.7.1.13	nppiAbs_32f_C3IR	218
7.7.1.14	nppiAbs_32f_C3R	219
7.7.1.15	nppiAbs_32f_C4IR	219
7.7.1.16	nppiAbs_32f_C4R	219

7.7.1.17	nppiAbsDiff_16u_C1R	220
7.7.1.18	nppiAbsDiff_32f_C1R	220
7.7.1.19	nppiAbsDiff_8u_C1R	220
7.7.1.20	nppiAbsDiffC_16u_C1R	221
7.7.1.21	nppiAbsDiffC_32f_C1R	221
7.7.1.22	nppiAbsDiffC_8u_C1R	222
7.7.1.23	nppiAdd_16s_AC4IRSfs	222
7.7.1.24	nppiAdd_16s_AC4RSfs	222
7.7.1.25	nppiAdd_16s_C1IRSfs	223
7.7.1.26	nppiAdd_16s_C1RSfs	223
7.7.1.27	nppiAdd_16s_C3IRSfs	224
7.7.1.28	nppiAdd_16s_C3RSfs	224
7.7.1.29	nppiAdd_16s_C4IRSfs	224
7.7.1.30	nppiAdd_16s_C4RSfs	225
7.7.1.31	nppiAdd_16sc_AC4IRSfs	225
7.7.1.32	nppiAdd_16sc_AC4RSfs	226
7.7.1.33	nppiAdd_16sc_C1IRSfs	226
7.7.1.34	nppiAdd_16sc_C1RSfs	226
7.7.1.35	nppiAdd_16sc_C3IRSfs	227
7.7.1.36	nppiAdd_16sc_C3RSfs	227
7.7.1.37	nppiAdd_16u_AC4IRSfs	228
7.7.1.38	nppiAdd_16u_AC4RSfs	228
7.7.1.39	nppiAdd_16u_C1IRSfs	229
7.7.1.40	nppiAdd_16u_C1RSfs	229
7.7.1.41	nppiAdd_16u_C3IRSfs	229
7.7.1.42	nppiAdd_16u_C3RSfs	230
7.7.1.43	nppiAdd_16u_C4IRSfs	230
7.7.1.44	nppiAdd_16u_C4RSfs	231
7.7.1.45	nppiAdd_32f_AC4IR	231
7.7.1.46	nppiAdd_32f_AC4R	231
7.7.1.47	nppiAdd_32f_C1IR	232
7.7.1.48	nppiAdd_32f_C1R	232
7.7.1.49	nppiAdd_32f_C3IR	233
7.7.1.50	nppiAdd_32f_C3R	233
7.7.1.51	nppiAdd_32f_C4IR	233
7.7.1.52	nppiAdd_32f_C4R	234

7.7.1.53	nppiAdd_32fc_AC4IR	234
7.7.1.54	nppiAdd_32fc_AC4R	234
7.7.1.55	nppiAdd_32fc_C1IR	235
7.7.1.56	nppiAdd_32fc_C1R	235
7.7.1.57	nppiAdd_32fc_C3IR	236
7.7.1.58	nppiAdd_32fc_C3R	236
7.7.1.59	nppiAdd_32fc_C4IR	236
7.7.1.60	nppiAdd_32fc_C4R	237
7.7.1.61	nppiAdd_32s_C1IRSfs	237
7.7.1.62	nppiAdd_32s_C1R	237
7.7.1.63	nppiAdd_32s_C1RSfs	238
7.7.1.64	nppiAdd_32s_C3IRSfs	238
7.7.1.65	nppiAdd_32s_C3RSfs	239
7.7.1.66	nppiAdd_32sc_AC4IRSfs	239
7.7.1.67	nppiAdd_32sc_AC4RSfs	239
7.7.1.68	nppiAdd_32sc_C1IRSfs	240
7.7.1.69	nppiAdd_32sc_C1RSfs	240
7.7.1.70	nppiAdd_32sc_C3IRSfs	241
7.7.1.71	nppiAdd_32sc_C3RSfs	241
7.7.1.72	nppiAdd_8u_AC4IRSfs	241
7.7.1.73	nppiAdd_8u_AC4RSfs	242
7.7.1.74	nppiAdd_8u_C1IRSfs	242
7.7.1.75	nppiAdd_8u_C1RSfs	243
7.7.1.76	nppiAdd_8u_C3IRSfs	243
7.7.1.77	nppiAdd_8u_C3RSfs	243
7.7.1.78	nppiAdd_8u_C4IRSfs	244
7.7.1.79	nppiAdd_8u_C4RSfs	244
7.7.1.80	nppiAddC_16s_AC4IRSfs	245
7.7.1.81	nppiAddC_16s_AC4RSfs	245
7.7.1.82	nppiAddC_16s_C1IRSfs	245
7.7.1.83	nppiAddC_16s_C1RSfs	246
7.7.1.84	nppiAddC_16s_C3IRSfs	246
7.7.1.85	nppiAddC_16s_C3RSfs	246
7.7.1.86	nppiAddC_16s_C4IRSfs	247
7.7.1.87	nppiAddC_16s_C4RSfs	247
7.7.1.88	nppiAddC_16sc_AC4IRSfs	248

7.7.1.89	nppiAddC_16sc_AC4RSfs	248
7.7.1.90	nppiAddC_16sc_C1IRSfs	248
7.7.1.91	nppiAddC_16sc_C1RSfs	249
7.7.1.92	nppiAddC_16sc_C3IRSfs	249
7.7.1.93	nppiAddC_16sc_C3RSfs	250
7.7.1.94	nppiAddC_16u_AC4IRSfs	250
7.7.1.95	nppiAddC_16u_AC4RSfs	250
7.7.1.96	nppiAddC_16u_C1IRSfs	251
7.7.1.97	nppiAddC_16u_C1RSfs	251
7.7.1.98	nppiAddC_16u_C3IRSfs	252
7.7.1.99	nppiAddC_16u_C3RSfs	252
7.7.1.100	nppiAddC_16u_C4IRSfs	252
7.7.1.101	nppiAddC_16u_C4RSfs	253
7.7.1.102	nppiAddC_32f_AC4IR	253
7.7.1.103	nppiAddC_32f_AC4R	253
7.7.1.104	nppiAddC_32f_C1IR	254
7.7.1.105	nppiAddC_32f_C1R	254
7.7.1.106	nppiAddC_32f_C3IR	254
7.7.1.107	nppiAddC_32f_C3R	255
7.7.1.108	nppiAddC_32f_C4IR	255
7.7.1.109	nppiAddC_32f_C4R	255
7.7.1.110	nppiAddC_32fc_AC4IR	256
7.7.1.111	nppiAddC_32fc_AC4R	256
7.7.1.112	nppiAddC_32fc_C1IR	256
7.7.1.113	nppiAddC_32fc_C1R	257
7.7.1.114	nppiAddC_32fc_C3IR	257
7.7.1.115	nppiAddC_32fc_C3R	257
7.7.1.116	nppiAddC_32fc_C4IR	258
7.7.1.117	nppiAddC_32fc_C4R	258
7.7.1.118	nppiAddC_32s_C1IRSfs	258
7.7.1.119	nppiAddC_32s_C1RSfs	259
7.7.1.120	nppiAddC_32s_C3IRSfs	259
7.7.1.121	nppiAddC_32s_C3RSfs	260
7.7.1.122	nppiAddC_32sc_AC4IRSfs	260
7.7.1.123	nppiAddC_32sc_AC4RSfs	260
7.7.1.124	nppiAddC_32sc_C1IRSfs	261

7.7.1.125 nppiAddC_32sc_C1RSfs	261
7.7.1.126 nppiAddC_32sc_C3IRSfs	262
7.7.1.127 nppiAddC_32sc_C3RSfs	262
7.7.1.128 nppiAddC_8u_AC4IRSfs	262
7.7.1.129 nppiAddC_8u_AC4RSfs	263
7.7.1.130 nppiAddC_8u_C1IRSfs	263
7.7.1.131 nppiAddC_8u_C1RSfs	264
7.7.1.132 nppiAddC_8u_C3IRSfs	264
7.7.1.133 nppiAddC_8u_C3RSfs	264
7.7.1.134 nppiAddC_8u_C4IRSfs	265
7.7.1.135 nppiAddC_8u_C4RSfs	265
7.7.1.136 nppiAddProduct_16u32f_C1IMR	266
7.7.1.137 nppiAddProduct_16u32f_C1IR	266
7.7.1.138 nppiAddProduct_32f_C1IMR	266
7.7.1.139 nppiAddProduct_32f_C1IR	267
7.7.1.140 nppiAddProduct_8u32f_C1IMR	267
7.7.1.141 nppiAddProduct_8u32f_C1IR	268
7.7.1.142 nppiAddSquare_16u32f_C1IMR	268
7.7.1.143 nppiAddSquare_16u32f_C1IR	269
7.7.1.144 nppiAddSquare_32f_C1IMR	269
7.7.1.145 nppiAddSquare_32f_C1IR	269
7.7.1.146 nppiAddSquare_8u32f_C1IMR	270
7.7.1.147 nppiAddSquare_8u32f_C1IR	270
7.7.1.148 nppiAddWeighted_16u32f_C1IMR	270
7.7.1.149 nppiAddWeighted_16u32f_C1IR	271
7.7.1.150 nppiAddWeighted_32f_C1IMR	271
7.7.1.151 nppiAddWeighted_32f_C1IR	272
7.7.1.152 nppiAddWeighted_8u32f_C1IMR	272
7.7.1.153 nppiAddWeighted_8u32f_C1IR	272
7.7.1.154 nppiAlphaComp_16s_AC1R	273
7.7.1.155 nppiAlphaComp_16u_AC1R	273
7.7.1.156 nppiAlphaComp_16u_AC4R	274
7.7.1.157 nppiAlphaComp_32f_AC1R	274
7.7.1.158 nppiAlphaComp_32f_AC4R	274
7.7.1.159 nppiAlphaComp_32s_AC1R	275
7.7.1.160 nppiAlphaComp_32s_AC4R	275

7.7.1.161 nppiAlphaComp_32u_AC1R	276
7.7.1.162 nppiAlphaComp_32u_AC4R	276
7.7.1.163 nppiAlphaComp_8s_AC1R	277
7.7.1.164 nppiAlphaComp_8u_AC1R	277
7.7.1.165 nppiAlphaComp_8u_AC4R	278
7.7.1.166 nppiAlphaCompC_16s_C1R	278
7.7.1.167 nppiAlphaCompC_16u_AC4R	279
7.7.1.168 nppiAlphaCompC_16u_C1R	279
7.7.1.169 nppiAlphaCompC_16u_C3R	280
7.7.1.170 nppiAlphaCompC_16u_C4R	280
7.7.1.171 nppiAlphaCompC_32f_C1R	281
7.7.1.172 nppiAlphaCompC_32s_C1R	281
7.7.1.173 nppiAlphaCompC_32u_C1R	282
7.7.1.174 nppiAlphaCompC_8s_C1R	282
7.7.1.175 nppiAlphaCompC_8u_AC4R	283
7.7.1.176 nppiAlphaCompC_8u_C1R	283
7.7.1.177 nppiAlphaCompC_8u_C3R	284
7.7.1.178 nppiAlphaCompC_8u_C4R	284
7.7.1.179 nppiAlphaPremul_16u_AC4IR	285
7.7.1.180 nppiAlphaPremul_16u_AC4R	285
7.7.1.181 nppiAlphaPremul_8u_AC4IR	285
7.7.1.182 nppiAlphaPremul_8u_AC4R	286
7.7.1.183 nppiAlphaPremulC_16u_AC4IR	286
7.7.1.184 nppiAlphaPremulC_16u_AC4R	286
7.7.1.185 nppiAlphaPremulC_16u_C1IR	287
7.7.1.186 nppiAlphaPremulC_16u_C1R	287
7.7.1.187 nppiAlphaPremulC_16u_C3IR	287
7.7.1.188 nppiAlphaPremulC_16u_C3R	288
7.7.1.189 nppiAlphaPremulC_16u_C4IR	288
7.7.1.190 nppiAlphaPremulC_16u_C4R	288
7.7.1.191 nppiAlphaPremulC_8u_AC4IR	289
7.7.1.192 nppiAlphaPremulC_8u_AC4R	289
7.7.1.193 nppiAlphaPremulC_8u_C1IR	289
7.7.1.194 nppiAlphaPremulC_8u_C1R	290
7.7.1.195 nppiAlphaPremulC_8u_C3IR	290
7.7.1.196 nppiAlphaPremulC_8u_C3R	290

7.7.1.197 nppiAlphaPremulC_8u_C4IR	291
7.7.1.198 nppiAlphaPremulC_8u_C4R	291
7.7.1.199 nppiAnd_16u_AC4IR	291
7.7.1.200 nppiAnd_16u_AC4R	292
7.7.1.201 nppiAnd_16u_C1IR	292
7.7.1.202 nppiAnd_16u_C1R	292
7.7.1.203 nppiAnd_16u_C3IR	293
7.7.1.204 nppiAnd_16u_C3R	293
7.7.1.205 nppiAnd_16u_C4IR	294
7.7.1.206 nppiAnd_16u_C4R	294
7.7.1.207 nppiAnd_32s_AC4IR	294
7.7.1.208 nppiAnd_32s_AC4R	295
7.7.1.209 nppiAnd_32s_C1IR	295
7.7.1.210 nppiAnd_32s_C1R	295
7.7.1.211 nppiAnd_32s_C3IR	296
7.7.1.212 nppiAnd_32s_C3R	296
7.7.1.213 nppiAnd_32s_C4IR	297
7.7.1.214 nppiAnd_32s_C4R	297
7.7.1.215 nppiAnd_8u_AC4IR	297
7.7.1.216 nppiAnd_8u_AC4R	298
7.7.1.217 nppiAnd_8u_C1IR	298
7.7.1.218 nppiAnd_8u_C1R	298
7.7.1.219 nppiAnd_8u_C3IR	299
7.7.1.220 nppiAnd_8u_C3R	299
7.7.1.221 nppiAnd_8u_C4IR	300
7.7.1.222 nppiAnd_8u_C4R	300
7.7.1.223 nppiAndC_16u_AC4IR	300
7.7.1.224 nppiAndC_16u_AC4R	301
7.7.1.225 nppiAndC_16u_C1IR	301
7.7.1.226 nppiAndC_16u_C1R	301
7.7.1.227 nppiAndC_16u_C3IR	302
7.7.1.228 nppiAndC_16u_C3R	302
7.7.1.229 nppiAndC_16u_C4IR	302
7.7.1.230 nppiAndC_16u_C4R	303
7.7.1.231 nppiAndC_32s_AC4IR	303
7.7.1.232 nppiAndC_32s_AC4R	303

7.7.1.233 nppiAndC_32s_C1IR	304
7.7.1.234 nppiAndC_32s_C1R	304
7.7.1.235 nppiAndC_32s_C3IR	304
7.7.1.236 nppiAndC_32s_C3R	305
7.7.1.237 nppiAndC_32s_C4IR	305
7.7.1.238 nppiAndC_32s_C4R	305
7.7.1.239 nppiAndC_8u_AC4IR	306
7.7.1.240 nppiAndC_8u_AC4R	306
7.7.1.241 nppiAndC_8u_C1IR	306
7.7.1.242 nppiAndC_8u_C1R	307
7.7.1.243 nppiAndC_8u_C3IR	307
7.7.1.244 nppiAndC_8u_C3R	307
7.7.1.245 nppiAndC_8u_C4IR	308
7.7.1.246 nppiAndC_8u_C4R	308
7.7.1.247 nppiDiv_16s_AC4IRSfs	308
7.7.1.248 nppiDiv_16s_AC4RSfs	309
7.7.1.249 nppiDiv_16s_C1IRSfs	309
7.7.1.250 nppiDiv_16s_C1RSfs	309
7.7.1.251 nppiDiv_16s_C3IRSfs	310
7.7.1.252 nppiDiv_16s_C3RSfs	310
7.7.1.253 nppiDiv_16s_C4IRSfs	311
7.7.1.254 nppiDiv_16s_C4RSfs	311
7.7.1.255 nppiDiv_16sc_AC4IRSfs	311
7.7.1.256 nppiDiv_16sc_AC4RSfs	312
7.7.1.257 nppiDiv_16sc_C1IRSfs	312
7.7.1.258 nppiDiv_16sc_C1RSfs	313
7.7.1.259 nppiDiv_16sc_C3IRSfs	313
7.7.1.260 nppiDiv_16sc_C3RSfs	313
7.7.1.261 nppiDiv_16u_AC4IRSfs	314
7.7.1.262 nppiDiv_16u_AC4RSfs	314
7.7.1.263 nppiDiv_16u_C1IRSfs	315
7.7.1.264 nppiDiv_16u_C1RSfs	315
7.7.1.265 nppiDiv_16u_C3IRSfs	316
7.7.1.266 nppiDiv_16u_C3RSfs	316
7.7.1.267 nppiDiv_16u_C4IRSfs	316
7.7.1.268 nppiDiv_16u_C4RSfs	317

7.7.1.269 nppiDiv_32f_AC4IR	317
7.7.1.270 nppiDiv_32f_AC4R	318
7.7.1.271 nppiDiv_32f_C1IR	318
7.7.1.272 nppiDiv_32f_C1R	318
7.7.1.273 nppiDiv_32f_C3IR	319
7.7.1.274 nppiDiv_32f_C3R	319
7.7.1.275 nppiDiv_32f_C4IR	320
7.7.1.276 nppiDiv_32f_C4R	320
7.7.1.277 nppiDiv_32fc_AC4IR	320
7.7.1.278 nppiDiv_32fc_AC4R	321
7.7.1.279 nppiDiv_32fc_C1IR	321
7.7.1.280 nppiDiv_32fc_C1R	321
7.7.1.281 nppiDiv_32fc_C3IR	322
7.7.1.282 nppiDiv_32fc_C3R	322
7.7.1.283 nppiDiv_32fc_C4IR	323
7.7.1.284 nppiDiv_32fc_C4R	323
7.7.1.285 nppiDiv_32s_C1IRSfs	323
7.7.1.286 nppiDiv_32s_C1R	324
7.7.1.287 nppiDiv_32s_C1RSfs	324
7.7.1.288 nppiDiv_32s_C3IRSfs	325
7.7.1.289 nppiDiv_32s_C3RSfs	325
7.7.1.290 nppiDiv_32sc_AC4IRSfs	325
7.7.1.291 nppiDiv_32sc_AC4RSfs	326
7.7.1.292 nppiDiv_32sc_C1IRSfs	326
7.7.1.293 nppiDiv_32sc_C1RSfs	327
7.7.1.294 nppiDiv_32sc_C3IRSfs	327
7.7.1.295 nppiDiv_32sc_C3RSfs	327
7.7.1.296 nppiDiv_8u_AC4IRSfs	328
7.7.1.297 nppiDiv_8u_AC4RSfs	328
7.7.1.298 nppiDiv_8u_C1IRSfs	329
7.7.1.299 nppiDiv_8u_C1RSfs	329
7.7.1.300 nppiDiv_8u_C3IRSfs	330
7.7.1.301 nppiDiv_8u_C3RSfs	330
7.7.1.302 nppiDiv_8u_C4IRSfs	330
7.7.1.303 nppiDiv_8u_C4RSfs	331
7.7.1.304 nppiDiv_Round_16s_AC4IRSfs	331

7.7.1.305 nppiDiv_Round_16s_AC4RSfs	332
7.7.1.306 nppiDiv_Round_16s_C1IRSfs	332
7.7.1.307 nppiDiv_Round_16s_C1RSfs	333
7.7.1.308 nppiDiv_Round_16s_C3IRSfs	333
7.7.1.309 nppiDiv_Round_16s_C3RSfs	334
7.7.1.310 nppiDiv_Round_16s_C4IRSfs	334
7.7.1.311 nppiDiv_Round_16s_C4RSfs	335
7.7.1.312 nppiDiv_Round_16u_AC4IRSfs	335
7.7.1.313 nppiDiv_Round_16u_AC4RSfs	336
7.7.1.314 nppiDiv_Round_16u_C1IRSfs	336
7.7.1.315 nppiDiv_Round_16u_C1RSfs	337
7.7.1.316 nppiDiv_Round_16u_C3IRSfs	337
7.7.1.317 nppiDiv_Round_16u_C3RSfs	338
7.7.1.318 nppiDiv_Round_16u_C4IRSfs	338
7.7.1.319 nppiDiv_Round_16u_C4RSfs	339
7.7.1.320 nppiDiv_Round_8u_AC4IRSfs	339
7.7.1.321 nppiDiv_Round_8u_AC4RSfs	340
7.7.1.322 nppiDiv_Round_8u_C1IRSfs	340
7.7.1.323 nppiDiv_Round_8u_C1RSfs	341
7.7.1.324 nppiDiv_Round_8u_C3IRSfs	341
7.7.1.325 nppiDiv_Round_8u_C3RSfs	342
7.7.1.326 nppiDiv_Round_8u_C4IRSfs	342
7.7.1.327 nppiDiv_Round_8u_C4RSfs	343
7.7.1.328 nppiDivC_16s_AC4IRSfs	343
7.7.1.329 nppiDivC_16s_AC4RSfs	343
7.7.1.330 nppiDivC_16s_C1IRSfs	344
7.7.1.331 nppiDivC_16s_C1RSfs	344
7.7.1.332 nppiDivC_16s_C3IRSfs	345
7.7.1.333 nppiDivC_16s_C3RSfs	345
7.7.1.334 nppiDivC_16s_C4IRSfs	345
7.7.1.335 nppiDivC_16s_C4RSfs	346
7.7.1.336 nppiDivC_16sc_AC4IRSfs	346
7.7.1.337 nppiDivC_16sc_AC4RSfs	346
7.7.1.338 nppiDivC_16sc_C1IRSfs	347
7.7.1.339 nppiDivC_16sc_C1RSfs	347
7.7.1.340 nppiDivC_16sc_C3IRSfs	348

7.7.1.341 nppiDivC_16sc_C3RSfs	348
7.7.1.342 nppiDivC_16u_AC4IRSfs	348
7.7.1.343 nppiDivC_16u_AC4RSfs	349
7.7.1.344 nppiDivC_16u_C1IRSfs	349
7.7.1.345 nppiDivC_16u_C1RSfs	350
7.7.1.346 nppiDivC_16u_C3IRSfs	350
7.7.1.347 nppiDivC_16u_C3RSfs	350
7.7.1.348 nppiDivC_16u_C4IRSfs	351
7.7.1.349 nppiDivC_16u_C4RSfs	351
7.7.1.350 nppiDivC_32f_AC4IR	352
7.7.1.351 nppiDivC_32f_AC4R	352
7.7.1.352 nppiDivC_32f_C1IR	352
7.7.1.353 nppiDivC_32f_C1R	353
7.7.1.354 nppiDivC_32f_C3IR	353
7.7.1.355 nppiDivC_32f_C3R	353
7.7.1.356 nppiDivC_32f_C4IR	354
7.7.1.357 nppiDivC_32f_C4R	354
7.7.1.358 nppiDivC_32fc_AC4IR	354
7.7.1.359 nppiDivC_32fc_AC4R	355
7.7.1.360 nppiDivC_32fc_C1IR	355
7.7.1.361 nppiDivC_32fc_C1R	355
7.7.1.362 nppiDivC_32fc_C3IR	356
7.7.1.363 nppiDivC_32fc_C3R	356
7.7.1.364 nppiDivC_32fc_C4IR	356
7.7.1.365 nppiDivC_32fc_C4R	357
7.7.1.366 nppiDivC_32s_C1IRSfs	357
7.7.1.367 nppiDivC_32s_C1RSfs	358
7.7.1.368 nppiDivC_32s_C3IRSfs	358
7.7.1.369 nppiDivC_32s_C3RSfs	358
7.7.1.370 nppiDivC_32sc_AC4IRSfs	359
7.7.1.371 nppiDivC_32sc_AC4RSfs	359
7.7.1.372 nppiDivC_32sc_C1IRSfs	360
7.7.1.373 nppiDivC_32sc_C1RSfs	360
7.7.1.374 nppiDivC_32sc_C3IRSfs	360
7.7.1.375 nppiDivC_32sc_C3RSfs	361
7.7.1.376 nppiDivC_8u_AC4IRSfs	361

7.7.1.377 nppiDivC_8u_AC4RSfs	362
7.7.1.378 nppiDivC_8u_C1IRSfs	362
7.7.1.379 nppiDivC_8u_C1RSfs	362
7.7.1.380 nppiDivC_8u_C3IRSfs	363
7.7.1.381 nppiDivC_8u_C3RSfs	363
7.7.1.382 nppiDivC_8u_C4IRSfs	364
7.7.1.383 nppiDivC_8u_C4RSfs	364
7.7.1.384 nppiExp_16s_C1IRSfs	364
7.7.1.385 nppiExp_16s_C1RSfs	365
7.7.1.386 nppiExp_16s_C3IRSfs	365
7.7.1.387 nppiExp_16s_C3RSfs	365
7.7.1.388 nppiExp_16u_C1IRSfs	366
7.7.1.389 nppiExp_16u_C1RSfs	366
7.7.1.390 nppiExp_16u_C3IRSfs	366
7.7.1.391 nppiExp_16u_C3RSfs	367
7.7.1.392 nppiExp_32f_C1IR	367
7.7.1.393 nppiExp_32f_C1R	367
7.7.1.394 nppiExp_32f_C3IR	368
7.7.1.395 nppiExp_32f_C3R	368
7.7.1.396 nppiExp_8u_C1IRSfs	368
7.7.1.397 nppiExp_8u_C1RSfs	369
7.7.1.398 nppiExp_8u_C3IRSfs	369
7.7.1.399 nppiExp_8u_C3RSfs	369
7.7.1.400 nppiLn_16s_C1IRSfs	370
7.7.1.401 nppiLn_16s_C1RSfs	370
7.7.1.402 nppiLn_16s_C3IRSfs	370
7.7.1.403 nppiLn_16s_C3RSfs	371
7.7.1.404 nppiLn_16u_C1IRSfs	371
7.7.1.405 nppiLn_16u_C1RSfs	371
7.7.1.406 nppiLn_16u_C3IRSfs	372
7.7.1.407 nppiLn_16u_C3RSfs	372
7.7.1.408 nppiLn_32f_C1IR	372
7.7.1.409 nppiLn_32f_C1R	373
7.7.1.410 nppiLn_32f_C3IR	373
7.7.1.411 nppiLn_32f_C3R	373
7.7.1.412 nppiLn_8u_C1IRSfs	374

7.7.1.413 nppiLn_8u_C1RSfs	374
7.7.1.414 nppiLn_8u_C3IRSfs	374
7.7.1.415 nppiLn_8u_C3RSfs	375
7.7.1.416 nppiLShiftC_16u_AC4IR	375
7.7.1.417 nppiLShiftC_16u_AC4R	375
7.7.1.418 nppiLShiftC_16u_C1IR	376
7.7.1.419 nppiLShiftC_16u_C1R	376
7.7.1.420 nppiLShiftC_16u_C3IR	376
7.7.1.421 nppiLShiftC_16u_C3R	377
7.7.1.422 nppiLShiftC_16u_C4IR	377
7.7.1.423 nppiLShiftC_16u_C4R	377
7.7.1.424 nppiLShiftC_32s_AC4IR	378
7.7.1.425 nppiLShiftC_32s_AC4R	378
7.7.1.426 nppiLShiftC_32s_C1IR	378
7.7.1.427 nppiLShiftC_32s_C1R	379
7.7.1.428 nppiLShiftC_32s_C3IR	379
7.7.1.429 nppiLShiftC_32s_C3R	379
7.7.1.430 nppiLShiftC_32s_C4IR	380
7.7.1.431 nppiLShiftC_32s_C4R	380
7.7.1.432 nppiLShiftC_8u_AC4IR	380
7.7.1.433 nppiLShiftC_8u_AC4R	381
7.7.1.434 nppiLShiftC_8u_C1IR	381
7.7.1.435 nppiLShiftC_8u_C1R	381
7.7.1.436 nppiLShiftC_8u_C3IR	382
7.7.1.437 nppiLShiftC_8u_C3R	382
7.7.1.438 nppiLShiftC_8u_C4IR	382
7.7.1.439 nppiLShiftC_8u_C4R	383
7.7.1.440 nppiMul_16s_AC4IRSfs	383
7.7.1.441 nppiMul_16s_AC4RSfs	383
7.7.1.442 nppiMul_16s_C1IRSfs	384
7.7.1.443 nppiMul_16s_C1RSfs	384
7.7.1.444 nppiMul_16s_C3IRSfs	385
7.7.1.445 nppiMul_16s_C3RSfs	385
7.7.1.446 nppiMul_16s_C4IRSfs	385
7.7.1.447 nppiMul_16s_C4RSfs	386
7.7.1.448 nppiMul_16sc_AC4IRSfs	386

7.7.1.449 nppiMul_16sc_AC4RSfs	387
7.7.1.450 nppiMul_16sc_C1IRSfs	387
7.7.1.451 nppiMul_16sc_C1RSfs	387
7.7.1.452 nppiMul_16sc_C3IRSfs	388
7.7.1.453 nppiMul_16sc_C3RSfs	388
7.7.1.454 nppiMul_16u_AC4IRSfs	389
7.7.1.455 nppiMul_16u_AC4RSfs	389
7.7.1.456 nppiMul_16u_C1IRSfs	390
7.7.1.457 nppiMul_16u_C1RSfs	390
7.7.1.458 nppiMul_16u_C3IRSfs	390
7.7.1.459 nppiMul_16u_C3RSfs	391
7.7.1.460 nppiMul_16u_C4IRSfs	391
7.7.1.461 nppiMul_16u_C4RSfs	392
7.7.1.462 nppiMul_32f_AC4IR	392
7.7.1.463 nppiMul_32f_AC4R	392
7.7.1.464 nppiMul_32f_C1IR	393
7.7.1.465 nppiMul_32f_C1R	393
7.7.1.466 nppiMul_32f_C3IR	394
7.7.1.467 nppiMul_32f_C3R	394
7.7.1.468 nppiMul_32f_C4IR	394
7.7.1.469 nppiMul_32f_C4R	395
7.7.1.470 nppiMul_32fc_AC4IR	395
7.7.1.471 nppiMul_32fc_AC4R	395
7.7.1.472 nppiMul_32fc_C1IR	396
7.7.1.473 nppiMul_32fc_C1R	396
7.7.1.474 nppiMul_32fc_C3IR	397
7.7.1.475 nppiMul_32fc_C3R	397
7.7.1.476 nppiMul_32fc_C4IR	397
7.7.1.477 nppiMul_32fc_C4R	398
7.7.1.478 nppiMul_32s_C1IRSfs	398
7.7.1.479 nppiMul_32s_C1R	399
7.7.1.480 nppiMul_32s_C1RSfs	399
7.7.1.481 nppiMul_32s_C3IRSfs	399
7.7.1.482 nppiMul_32s_C3RSfs	400
7.7.1.483 nppiMul_32sc_AC4IRSfs	400
7.7.1.484 nppiMul_32sc_AC4RSfs	401

7.7.1.485 nppiMul_32sc_C1IRSfs	401
7.7.1.486 nppiMul_32sc_C1RSfs	401
7.7.1.487 nppiMul_32sc_C3IRSfs	402
7.7.1.488 nppiMul_32sc_C3RSfs	402
7.7.1.489 nppiMul_8u_AC4IRSfs	403
7.7.1.490 nppiMul_8u_AC4RSfs	403
7.7.1.491 nppiMul_8u_C1IRSfs	404
7.7.1.492 nppiMul_8u_C1RSfs	404
7.7.1.493 nppiMul_8u_C3IRSfs	404
7.7.1.494 nppiMul_8u_C3RSfs	405
7.7.1.495 nppiMul_8u_C4IRSfs	405
7.7.1.496 nppiMul_8u_C4RSfs	406
7.7.1.497 nppiMulC_16s_AC4IRSfs	406
7.7.1.498 nppiMulC_16s_AC4RSfs	406
7.7.1.499 nppiMulC_16s_C1IRSfs	407
7.7.1.500 nppiMulC_16s_C1RSfs	407
7.7.1.501 nppiMulC_16s_C3IRSfs	408
7.7.1.502 nppiMulC_16s_C3RSfs	408
7.7.1.503 nppiMulC_16s_C4IRSfs	408
7.7.1.504 nppiMulC_16s_C4RSfs	409
7.7.1.505 nppiMulC_16sc_AC4IRSfs	409
7.7.1.506 nppiMulC_16sc_AC4RSfs	409
7.7.1.507 nppiMulC_16sc_C1IRSfs	410
7.7.1.508 nppiMulC_16sc_C1RSfs	410
7.7.1.509 nppiMulC_16sc_C3IRSfs	411
7.7.1.510 nppiMulC_16sc_C3RSfs	411
7.7.1.511 nppiMulC_16u_AC4IRSfs	411
7.7.1.512 nppiMulC_16u_AC4RSfs	412
7.7.1.513 nppiMulC_16u_C1IRSfs	412
7.7.1.514 nppiMulC_16u_C1RSfs	413
7.7.1.515 nppiMulC_16u_C3IRSfs	413
7.7.1.516 nppiMulC_16u_C3RSfs	413
7.7.1.517 nppiMulC_16u_C4IRSfs	414
7.7.1.518 nppiMulC_16u_C4RSfs	414
7.7.1.519 nppiMulC_32f_AC4IR	415
7.7.1.520 nppiMulC_32f_AC4R	415

7.7.1.521 nppiMulC_32f_C1IR	415
7.7.1.522 nppiMulC_32f_C1R	416
7.7.1.523 nppiMulC_32f_C3IR	416
7.7.1.524 nppiMulC_32f_C3R	416
7.7.1.525 nppiMulC_32f_C4IR	417
7.7.1.526 nppiMulC_32f_C4R	417
7.7.1.527 nppiMulC_32fc_AC4IR	417
7.7.1.528 nppiMulC_32fc_AC4R	418
7.7.1.529 nppiMulC_32fc_C1IR	418
7.7.1.530 nppiMulC_32fc_C1R	418
7.7.1.531 nppiMulC_32fc_C3IR	419
7.7.1.532 nppiMulC_32fc_C3R	419
7.7.1.533 nppiMulC_32fc_C4IR	419
7.7.1.534 nppiMulC_32fc_C4R	420
7.7.1.535 nppiMulC_32s_C1IRSfs	420
7.7.1.536 nppiMulC_32s_C1RSfs	421
7.7.1.537 nppiMulC_32s_C3IRSfs	421
7.7.1.538 nppiMulC_32s_C3RSfs	421
7.7.1.539 nppiMulC_32sc_AC4IRSfs	422
7.7.1.540 nppiMulC_32sc_AC4RSfs	422
7.7.1.541 nppiMulC_32sc_C1IRSfs	423
7.7.1.542 nppiMulC_32sc_C1RSfs	423
7.7.1.543 nppiMulC_32sc_C3IRSfs	423
7.7.1.544 nppiMulC_32sc_C3RSfs	424
7.7.1.545 nppiMulC_8u_AC4IRSfs	424
7.7.1.546 nppiMulC_8u_AC4RSfs	425
7.7.1.547 nppiMulC_8u_C1IRSfs	425
7.7.1.548 nppiMulC_8u_C1RSfs	425
7.7.1.549 nppiMulC_8u_C3IRSfs	426
7.7.1.550 nppiMulC_8u_C3RSfs	426
7.7.1.551 nppiMulC_8u_C4IRSfs	427
7.7.1.552 nppiMulC_8u_C4RSfs	427
7.7.1.553 nppiMulCScale_16u_AC4IR	427
7.7.1.554 nppiMulCScale_16u_AC4R	428
7.7.1.555 nppiMulCScale_16u_C1IR	428
7.7.1.556 nppiMulCScale_16u_C1R	428

7.7.1.557 nppiMulCScale_16u_C3IR	429
7.7.1.558 nppiMulCScale_16u_C3R	429
7.7.1.559 nppiMulCScale_16u_C4IR	429
7.7.1.560 nppiMulCScale_16u_C4R	430
7.7.1.561 nppiMulCScale_8u_AC4IR	430
7.7.1.562 nppiMulCScale_8u_AC4R	430
7.7.1.563 nppiMulCScale_8u_C1IR	431
7.7.1.564 nppiMulCScale_8u_C1R	431
7.7.1.565 nppiMulCScale_8u_C3IR	431
7.7.1.566 nppiMulCScale_8u_C3R	432
7.7.1.567 nppiMulCScale_8u_C4IR	432
7.7.1.568 nppiMulCScale_8u_C4R	432
7.7.1.569 nppiMulScale_16u_AC4IR	433
7.7.1.570 nppiMulScale_16u_AC4R	433
7.7.1.571 nppiMulScale_16u_C1IR	433
7.7.1.572 nppiMulScale_16u_C1R	434
7.7.1.573 nppiMulScale_16u_C3IR	434
7.7.1.574 nppiMulScale_16u_C3R	435
7.7.1.575 nppiMulScale_16u_C4IR	435
7.7.1.576 nppiMulScale_16u_C4R	435
7.7.1.577 nppiMulScale_8u_AC4IR	436
7.7.1.578 nppiMulScale_8u_AC4R	436
7.7.1.579 nppiMulScale_8u_C1IR	437
7.7.1.580 nppiMulScale_8u_C1R	437
7.7.1.581 nppiMulScale_8u_C3IR	437
7.7.1.582 nppiMulScale_8u_C3R	438
7.7.1.583 nppiMulScale_8u_C4IR	438
7.7.1.584 nppiMulScale_8u_C4R	438
7.7.1.585 nppiNot_8u_AC4IR	439
7.7.1.586 nppiNot_8u_AC4R	439
7.7.1.587 nppiNot_8u_C1IR	440
7.7.1.588 nppiNot_8u_C1R	440
7.7.1.589 nppiNot_8u_C3IR	440
7.7.1.590 nppiNot_8u_C3R	441
7.7.1.591 nppiNot_8u_C4IR	441
7.7.1.592 nppiNot_8u_C4R	441

7.7.1.593 nppiOr_16u_AC4IR	442
7.7.1.594 nppiOr_16u_AC4R	442
7.7.1.595 nppiOr_16u_C1IR	442
7.7.1.596 nppiOr_16u_C1R	443
7.7.1.597 nppiOr_16u_C3IR	443
7.7.1.598 nppiOr_16u_C3R	443
7.7.1.599 nppiOr_16u_C4IR	444
7.7.1.600 nppiOr_16u_C4R	444
7.7.1.601 nppiOr_32s_AC4IR	445
7.7.1.602 nppiOr_32s_AC4R	445
7.7.1.603 nppiOr_32s_C1IR	445
7.7.1.604 nppiOr_32s_C1R	446
7.7.1.605 nppiOr_32s_C3IR	446
7.7.1.606 nppiOr_32s_C3R	446
7.7.1.607 nppiOr_32s_C4IR	447
7.7.1.608 nppiOr_32s_C4R	447
7.7.1.609 nppiOr_8u_AC4IR	448
7.7.1.610 nppiOr_8u_AC4R	448
7.7.1.611 nppiOr_8u_C1IR	448
7.7.1.612 nppiOr_8u_C1R	449
7.7.1.613 nppiOr_8u_C3IR	449
7.7.1.614 nppiOr_8u_C3R	449
7.7.1.615 nppiOr_8u_C4IR	450
7.7.1.616 nppiOr_8u_C4R	450
7.7.1.617 nppiOrC_16u_AC4IR	451
7.7.1.618 nppiOrC_16u_AC4R	451
7.7.1.619 nppiOrC_16u_C1IR	451
7.7.1.620 nppiOrC_16u_C1R	452
7.7.1.621 nppiOrC_16u_C3IR	452
7.7.1.622 nppiOrC_16u_C3R	452
7.7.1.623 nppiOrC_16u_C4IR	453
7.7.1.624 nppiOrC_16u_C4R	453
7.7.1.625 nppiOrC_32s_AC4IR	453
7.7.1.626 nppiOrC_32s_AC4R	454
7.7.1.627 nppiOrC_32s_C1IR	454
7.7.1.628 nppiOrC_32s_C1R	454

7.7.1.629 nppiOrC_32s_C3IR	455
7.7.1.630 nppiOrC_32s_C3R	455
7.7.1.631 nppiOrC_32s_C4IR	455
7.7.1.632 nppiOrC_32s_C4R	456
7.7.1.633 nppiOrC_8u_AC4IR	456
7.7.1.634 nppiOrC_8u_AC4R	456
7.7.1.635 nppiOrC_8u_C1IR	457
7.7.1.636 nppiOrC_8u_C1R	457
7.7.1.637 nppiOrC_8u_C3IR	457
7.7.1.638 nppiOrC_8u_C3R	458
7.7.1.639 nppiOrC_8u_C4IR	458
7.7.1.640 nppiOrC_8u_C4R	458
7.7.1.641 nppiRShiftC_16s_AC4IR	459
7.7.1.642 nppiRShiftC_16s_AC4R	459
7.7.1.643 nppiRShiftC_16s_C1IR	459
7.7.1.644 nppiRShiftC_16s_C1R	460
7.7.1.645 nppiRShiftC_16s_C3IR	460
7.7.1.646 nppiRShiftC_16s_C3R	460
7.7.1.647 nppiRShiftC_16s_C4IR	461
7.7.1.648 nppiRShiftC_16s_C4R	461
7.7.1.649 nppiRShiftC_16u_AC4IR	461
7.7.1.650 nppiRShiftC_16u_AC4R	462
7.7.1.651 nppiRShiftC_16u_C1IR	462
7.7.1.652 nppiRShiftC_16u_C1R	462
7.7.1.653 nppiRShiftC_16u_C3IR	463
7.7.1.654 nppiRShiftC_16u_C3R	463
7.7.1.655 nppiRShiftC_16u_C4IR	463
7.7.1.656 nppiRShiftC_16u_C4R	464
7.7.1.657 nppiRShiftC_32s_AC4IR	464
7.7.1.658 nppiRShiftC_32s_AC4R	464
7.7.1.659 nppiRShiftC_32s_C1IR	465
7.7.1.660 nppiRShiftC_32s_C1R	465
7.7.1.661 nppiRShiftC_32s_C3IR	465
7.7.1.662 nppiRShiftC_32s_C3R	466
7.7.1.663 nppiRShiftC_32s_C4IR	466
7.7.1.664 nppiRShiftC_32s_C4R	466

7.7.1.665 nppiRShiftC_8s_AC4IR	467
7.7.1.666 nppiRShiftC_8s_AC4R	467
7.7.1.667 nppiRShiftC_8s_C1IR	467
7.7.1.668 nppiRShiftC_8s_C1R	468
7.7.1.669 nppiRShiftC_8s_C3IR	468
7.7.1.670 nppiRShiftC_8s_C3R	468
7.7.1.671 nppiRShiftC_8s_C4IR	469
7.7.1.672 nppiRShiftC_8s_C4R	469
7.7.1.673 nppiRShiftC_8u_AC4IR	469
7.7.1.674 nppiRShiftC_8u_AC4R	470
7.7.1.675 nppiRShiftC_8u_C1IR	470
7.7.1.676 nppiRShiftC_8u_C1R	470
7.7.1.677 nppiRShiftC_8u_C3IR	471
7.7.1.678 nppiRShiftC_8u_C3R	471
7.7.1.679 nppiRShiftC_8u_C4IR	471
7.7.1.680 nppiRShiftC_8u_C4R	472
7.7.1.681 nppiSqr_16s_AC4IRSfs	472
7.7.1.682 nppiSqr_16s_AC4RSfs	472
7.7.1.683 nppiSqr_16s_C1IRSfs	473
7.7.1.684 nppiSqr_16s_C1RSfs	473
7.7.1.685 nppiSqr_16s_C3IRSfs	473
7.7.1.686 nppiSqr_16s_C3RSfs	474
7.7.1.687 nppiSqr_16s_C4IRSfs	474
7.7.1.688 nppiSqr_16s_C4RSfs	474
7.7.1.689 nppiSqr_16u_AC4IRSfs	475
7.7.1.690 nppiSqr_16u_AC4RSfs	475
7.7.1.691 nppiSqr_16u_C1IRSfs	475
7.7.1.692 nppiSqr_16u_C1RSfs	476
7.7.1.693 nppiSqr_16u_C3IRSfs	476
7.7.1.694 nppiSqr_16u_C3RSfs	476
7.7.1.695 nppiSqr_16u_C4IRSfs	477
7.7.1.696 nppiSqr_16u_C4RSfs	477
7.7.1.697 nppiSqr_32f_AC4IR	477
7.7.1.698 nppiSqr_32f_AC4R	478
7.7.1.699 nppiSqr_32f_C1IR	478
7.7.1.700 nppiSqr_32f_C1R	478

7.7.1.701 nppiSqr_32f_C3IR	479
7.7.1.702 nppiSqr_32f_C3R	479
7.7.1.703 nppiSqr_32f_C4IR	479
7.7.1.704 nppiSqr_32f_C4R	480
7.7.1.705 nppiSqr_8u_AC4IRSfs	480
7.7.1.706 nppiSqr_8u_AC4RSfs	480
7.7.1.707 nppiSqr_8u_C1IRSfs	481
7.7.1.708 nppiSqr_8u_C1RSfs	481
7.7.1.709 nppiSqr_8u_C3IRSfs	481
7.7.1.710 nppiSqr_8u_C3RSfs	482
7.7.1.711 nppiSqr_8u_C4IRSfs	482
7.7.1.712 nppiSqr_8u_C4RSfs	482
7.7.1.713 nppiSqrt_16s_AC4IRSfs	483
7.7.1.714 nppiSqrt_16s_AC4RSfs	483
7.7.1.715 nppiSqrt_16s_C1IRSfs	483
7.7.1.716 nppiSqrt_16s_C1RSfs	484
7.7.1.717 nppiSqrt_16s_C3IRSfs	484
7.7.1.718 nppiSqrt_16s_C3RSfs	484
7.7.1.719 nppiSqrt_16u_AC4IRSfs	485
7.7.1.720 nppiSqrt_16u_AC4RSfs	485
7.7.1.721 nppiSqrt_16u_C1IRSfs	485
7.7.1.722 nppiSqrt_16u_C1RSfs	486
7.7.1.723 nppiSqrt_16u_C3IRSfs	486
7.7.1.724 nppiSqrt_16u_C3RSfs	486
7.7.1.725 nppiSqrt_32f_AC4IR	487
7.7.1.726 nppiSqrt_32f_AC4R	487
7.7.1.727 nppiSqrt_32f_C1IR	487
7.7.1.728 nppiSqrt_32f_C1R	488
7.7.1.729 nppiSqrt_32f_C3IR	488
7.7.1.730 nppiSqrt_32f_C3R	488
7.7.1.731 nppiSqrt_32f_C4IR	489
7.7.1.732 nppiSqrt_32f_C4R	489
7.7.1.733 nppiSqrt_8u_AC4IRSfs	489
7.7.1.734 nppiSqrt_8u_AC4RSfs	490
7.7.1.735 nppiSqrt_8u_C1IRSfs	490
7.7.1.736 nppiSqrt_8u_C1RSfs	490

7.7.1.737 nppiSqrt_8u_C3IRSfs	491
7.7.1.738 nppiSqrt_8u_C3RSfs	491
7.7.1.739 nppiSub_16s_AC4IRSfs	491
7.7.1.740 nppiSub_16s_AC4RSfs	492
7.7.1.741 nppiSub_16s_C1IRSfs	492
7.7.1.742 nppiSub_16s_C1RSfs	493
7.7.1.743 nppiSub_16s_C3IRSfs	493
7.7.1.744 nppiSub_16s_C3RSfs	493
7.7.1.745 nppiSub_16s_C4IRSfs	494
7.7.1.746 nppiSub_16s_C4RSfs	494
7.7.1.747 nppiSub_16sc_AC4IRSfs	495
7.7.1.748 nppiSub_16sc_AC4RSfs	495
7.7.1.749 nppiSub_16sc_C1IRSfs	496
7.7.1.750 nppiSub_16sc_C1RSfs	496
7.7.1.751 nppiSub_16sc_C3IRSfs	496
7.7.1.752 nppiSub_16sc_C3RSfs	497
7.7.1.753 nppiSub_16u_AC4IRSfs	497
7.7.1.754 nppiSub_16u_AC4RSfs	498
7.7.1.755 nppiSub_16u_C1IRSfs	498
7.7.1.756 nppiSub_16u_C1RSfs	498
7.7.1.757 nppiSub_16u_C3IRSfs	499
7.7.1.758 nppiSub_16u_C3RSfs	499
7.7.1.759 nppiSub_16u_C4IRSfs	500
7.7.1.760 nppiSub_16u_C4RSfs	500
7.7.1.761 nppiSub_32f_AC4IR	501
7.7.1.762 nppiSub_32f_AC4R	501
7.7.1.763 nppiSub_32f_C1IR	501
7.7.1.764 nppiSub_32f_C1R	502
7.7.1.765 nppiSub_32f_C3IR	502
7.7.1.766 nppiSub_32f_C3R	502
7.7.1.767 nppiSub_32f_C4IR	503
7.7.1.768 nppiSub_32f_C4R	503
7.7.1.769 nppiSub_32fc_AC4IR	504
7.7.1.770 nppiSub_32fc_AC4R	504
7.7.1.771 nppiSub_32fc_C1IR	504
7.7.1.772 nppiSub_32fc_C1R	505

7.7.1.773 nppiSub_32fc_C3IR	505
7.7.1.774 nppiSub_32fc_C3R	506
7.7.1.775 nppiSub_32fc_C4IR	506
7.7.1.776 nppiSub_32fc_C4R	506
7.7.1.777 nppiSub_32s_C1IRSfs	507
7.7.1.778 nppiSub_32s_C1R	507
7.7.1.779 nppiSub_32s_C1RSfs	508
7.7.1.780 nppiSub_32s_C3IRSfs	508
7.7.1.781 nppiSub_32s_C3RSfs	508
7.7.1.782 nppiSub_32sc_AC4IRSfs	509
7.7.1.783 nppiSub_32sc_AC4RSfs	509
7.7.1.784 nppiSub_32sc_C1IRSfs	510
7.7.1.785 nppiSub_32sc_C1RSfs	510
7.7.1.786 nppiSub_32sc_C3IRSfs	511
7.7.1.787 nppiSub_32sc_C3RSfs	511
7.7.1.788 nppiSub_8u_AC4IRSfs	511
7.7.1.789 nppiSub_8u_AC4RSfs	512
7.7.1.790 nppiSub_8u_C1IRSfs	512
7.7.1.791 nppiSub_8u_C1RSfs	513
7.7.1.792 nppiSub_8u_C3IRSfs	513
7.7.1.793 nppiSub_8u_C3RSfs	513
7.7.1.794 nppiSub_8u_C4IRSfs	514
7.7.1.795 nppiSub_8u_C4RSfs	514
7.7.1.796 nppiSubC_16s_AC4IRSfs	515
7.7.1.797 nppiSubC_16s_AC4RSfs	515
7.7.1.798 nppiSubC_16s_C1IRSfs	515
7.7.1.799 nppiSubC_16s_C1RSfs	516
7.7.1.800 nppiSubC_16s_C3IRSfs	516
7.7.1.801 nppiSubC_16s_C3RSfs	516
7.7.1.802 nppiSubC_16s_C4IRSfs	517
7.7.1.803 nppiSubC_16s_C4RSfs	517
7.7.1.804 nppiSubC_16sc_AC4IRSfs	518
7.7.1.805 nppiSubC_16sc_AC4RSfs	518
7.7.1.806 nppiSubC_16sc_C1IRSfs	518
7.7.1.807 nppiSubC_16sc_C1RSfs	519
7.7.1.808 nppiSubC_16sc_C3IRSfs	519

7.7.1.809 nppiSubC_16sc_C3RSfs	520
7.7.1.810 nppiSubC_16u_AC4IRSfs	520
7.7.1.811 nppiSubC_16u_AC4RSfs	520
7.7.1.812 nppiSubC_16u_C1IRSfs	521
7.7.1.813 nppiSubC_16u_C1RSfs	521
7.7.1.814 nppiSubC_16u_C3IRSfs	522
7.7.1.815 nppiSubC_16u_C3RSfs	522
7.7.1.816 nppiSubC_16u_C4IRSfs	522
7.7.1.817 nppiSubC_16u_C4RSfs	523
7.7.1.818 nppiSubC_32f_AC4IR	523
7.7.1.819 nppiSubC_32f_AC4R	523
7.7.1.820 nppiSubC_32f_C1IR	524
7.7.1.821 nppiSubC_32f_C1R	524
7.7.1.822 nppiSubC_32f_C3IR	524
7.7.1.823 nppiSubC_32f_C3R	525
7.7.1.824 nppiSubC_32f_C4IR	525
7.7.1.825 nppiSubC_32f_C4R	525
7.7.1.826 nppiSubC_32fc_AC4IR	526
7.7.1.827 nppiSubC_32fc_AC4R	526
7.7.1.828 nppiSubC_32fc_C1IR	526
7.7.1.829 nppiSubC_32fc_C1R	527
7.7.1.830 nppiSubC_32fc_C3IR	527
7.7.1.831 nppiSubC_32fc_C3R	527
7.7.1.832 nppiSubC_32fc_C4IR	528
7.7.1.833 nppiSubC_32fc_C4R	528
7.7.1.834 nppiSubC_32s_C1IRSfs	528
7.7.1.835 nppiSubC_32s_C1RSfs	529
7.7.1.836 nppiSubC_32s_C3IRSfs	529
7.7.1.837 nppiSubC_32s_C3RSfs	530
7.7.1.838 nppiSubC_32sc_AC4IRSfs	530
7.7.1.839 nppiSubC_32sc_AC4RSfs	530
7.7.1.840 nppiSubC_32sc_C1IRSfs	531
7.7.1.841 nppiSubC_32sc_C1RSfs	531
7.7.1.842 nppiSubC_32sc_C3IRSfs	532
7.7.1.843 nppiSubC_32sc_C3RSfs	532
7.7.1.844 nppiSubC_8u_AC4IRSfs	532

7.7.1.845 nppiSubC_8u_AC4RSfs	533
7.7.1.846 nppiSubC_8u_C1IRSfs	533
7.7.1.847 nppiSubC_8u_C1RSfs	534
7.7.1.848 nppiSubC_8u_C3IRSfs	534
7.7.1.849 nppiSubC_8u_C3RSfs	534
7.7.1.850 nppiSubC_8u_C4IRSfs	535
7.7.1.851 nppiSubC_8u_C4RSfs	535
7.7.1.852 nppiXor_16u_AC4IR	536
7.7.1.853 nppiXor_16u_AC4R	536
7.7.1.854 nppiXor_16u_C1IR	536
7.7.1.855 nppiXor_16u_C1R	537
7.7.1.856 nppiXor_16u_C3IR	537
7.7.1.857 nppiXor_16u_C3R	537
7.7.1.858 nppiXor_16u_C4IR	538
7.7.1.859 nppiXor_16u_C4R	538
7.7.1.860 nppiXor_32s_AC4IR	539
7.7.1.861 nppiXor_32s_AC4R	539
7.7.1.862 nppiXor_32s_C1IR	539
7.7.1.863 nppiXor_32s_C1R	540
7.7.1.864 nppiXor_32s_C3IR	540
7.7.1.865 nppiXor_32s_C3R	540
7.7.1.866 nppiXor_32s_C4IR	541
7.7.1.867 nppiXor_32s_C4R	541
7.7.1.868 nppiXor_8u_AC4IR	542
7.7.1.869 nppiXor_8u_AC4R	542
7.7.1.870 nppiXor_8u_C1IR	542
7.7.1.871 nppiXor_8u_C1R	543
7.7.1.872 nppiXor_8u_C3IR	543
7.7.1.873 nppiXor_8u_C3R	543
7.7.1.874 nppiXor_8u_C4IR	544
7.7.1.875 nppiXor_8u_C4R	544
7.7.1.876 nppiXorC_16u_AC4IR	545
7.7.1.877 nppiXorC_16u_AC4R	545
7.7.1.878 nppiXorC_16u_C1IR	545
7.7.1.879 nppiXorC_16u_C1R	546
7.7.1.880 nppiXorC_16u_C3IR	546

7.7.1.881	<code>nppiXorC_16u_C3R</code>	546
7.7.1.882	<code>nppiXorC_16u_C4IR</code>	547
7.7.1.883	<code>nppiXorC_16u_C4R</code>	547
7.7.1.884	<code>nppiXorC_32s_AC4IR</code>	547
7.7.1.885	<code>nppiXorC_32s_AC4R</code>	548
7.7.1.886	<code>nppiXorC_32s_C1IR</code>	548
7.7.1.887	<code>nppiXorC_32s_C1R</code>	548
7.7.1.888	<code>nppiXorC_32s_C3IR</code>	549
7.7.1.889	<code>nppiXorC_32s_C3R</code>	549
7.7.1.890	<code>nppiXorC_32s_C4IR</code>	549
7.7.1.891	<code>nppiXorC_32s_C4R</code>	550
7.7.1.892	<code>nppiXorC_8u_AC4IR</code>	550
7.7.1.893	<code>nppiXorC_8u_AC4R</code>	550
7.7.1.894	<code>nppiXorC_8u_C1IR</code>	551
7.7.1.895	<code>nppiXorC_8u_C1R</code>	551
7.7.1.896	<code>nppiXorC_8u_C3IR</code>	551
7.7.1.897	<code>nppiXorC_8u_C3R</code>	552
7.7.1.898	<code>nppiXorC_8u_C4IR</code>	552
7.7.1.899	<code>nppiXorC_8u_C4R</code>	552
7.8	Threshold and Compare Operations	553
7.8.1	Detailed Description	553
7.8.2	Function Documentation	553
7.8.2.1	<code>nppiCompare_32f_C1R</code>	553
7.8.2.2	<code>nppiCompare_8u_AC4R</code>	554
7.8.2.3	<code>nppiCompare_8u_C4R</code>	554
7.8.2.4	<code>nppiThreshold_32f_C1R</code>	555
7.8.2.5	<code>nppiThreshold_8u_AC4R</code>	555
7.9	Statistics Functions	557
7.9.1	Detailed Description	562
7.9.2	Function Documentation	562
7.9.2.1	<code>nppiEvenLevelsHost_32s</code>	562
7.9.2.2	<code>nppiHistogramEven_16s_AC4R</code>	563
7.9.2.3	<code>nppiHistogramEven_16s_C1R</code>	563
7.9.2.4	<code>nppiHistogramEven_16s_C4R</code>	564
7.9.2.5	<code>nppiHistogramEven_16u_AC4R</code>	564
7.9.2.6	<code>nppiHistogramEven_16u_C1R</code>	565

7.9.2.7	nppiHistogramEven_16u_C4R	565
7.9.2.8	nppiHistogramEven_8u_AC4R	565
7.9.2.9	nppiHistogramEven_8u_C1R	566
7.9.2.10	nppiHistogramEven_8u_C4R	566
7.9.2.11	nppiHistogramEvenGetBufferSize_16s_AC4R	567
7.9.2.12	nppiHistogramEvenGetBufferSize_16s_C1R	567
7.9.2.13	nppiHistogramEvenGetBufferSize_16s_C4R	567
7.9.2.14	nppiHistogramEvenGetBufferSize_16u_AC4R	568
7.9.2.15	nppiHistogramEvenGetBufferSize_16u_C1R	568
7.9.2.16	nppiHistogramEvenGetBufferSize_16u_C4R	568
7.9.2.17	nppiHistogramEvenGetBufferSize_8u_AC4R	569
7.9.2.18	nppiHistogramEvenGetBufferSize_8u_C1R	569
7.9.2.19	nppiHistogramEvenGetBufferSize_8u_C4R	569
7.9.2.20	nppiHistogramRange_16s_AC4R	569
7.9.2.21	nppiHistogramRange_16s_C1R	570
7.9.2.22	nppiHistogramRange_16s_C4R	570
7.9.2.23	nppiHistogramRange_16u_AC4R	571
7.9.2.24	nppiHistogramRange_16u_C1R	571
7.9.2.25	nppiHistogramRange_16u_C4R	572
7.9.2.26	nppiHistogramRange_32f_AC4R	572
7.9.2.27	nppiHistogramRange_32f_C1R	573
7.9.2.28	nppiHistogramRange_32f_C4R	573
7.9.2.29	nppiHistogramRange_8u_AC4R	573
7.9.2.30	nppiHistogramRange_8u_C1R	574
7.9.2.31	nppiHistogramRange_8u_C4R	574
7.9.2.32	nppiHistogramRangeGetBufferSize_16s_AC4R	575
7.9.2.33	nppiHistogramRangeGetBufferSize_16s_C1R	575
7.9.2.34	nppiHistogramRangeGetBufferSize_16s_C4R	575
7.9.2.35	nppiHistogramRangeGetBufferSize_16u_AC4R	576
7.9.2.36	nppiHistogramRangeGetBufferSize_16u_C1R	576
7.9.2.37	nppiHistogramRangeGetBufferSize_16u_C4R	576
7.9.2.38	nppiHistogramRangeGetBufferSize_32f_AC4R	576
7.9.2.39	nppiHistogramRangeGetBufferSize_32f_C1R	577
7.9.2.40	nppiHistogramRangeGetBufferSize_32f_C4R	577
7.9.2.41	nppiHistogramRangeGetBufferSize_8u_AC4R	577
7.9.2.42	nppiHistogramRangeGetBufferSize_8u_C1R	578

7.9.2.43	nppiHistogramRangeGetBufferSize_8u_C4R	578
7.9.2.44	nppiMean_StdDev_8u_C1R	578
7.9.2.45	nppiMeanStdDev8uC1RGetBufferHostSize	579
7.9.2.46	nppiMinMax_8u_C1R	579
7.9.2.47	nppiMinMax_8u_C4R	579
7.9.2.48	nppiMinMaxGetBufferSize_8u_C1R	580
7.9.2.49	nppiMinMaxGetBufferSize_8u_C4R	580
7.9.2.50	nppiNormDiff_Inf_8u_C1R	580
7.9.2.51	nppiNormDiff_L1_8u_C1R	581
7.9.2.52	nppiNormDiff_L2_8u_C1R	581
7.9.2.53	nppiRectStdDev_32s32f_C1R	581
7.9.2.54	nppiReductionGetBufferHostSize_8u_C1R	582
7.9.2.55	nppiReductionGetBufferHostSize_8u_C4R	582
7.9.2.56	nppiSqrIntegral_8u32s32f_C1R	582
7.9.2.57	nppiSum_8u64s_C1R	583
7.9.2.58	nppiSum_8u64s_C4R	583
7.9.2.59	nppiSum_8u_C1R	584
7.9.2.60	nppiSum_8u_C4R	584
7.10	Filtering Functions	585
7.10.1	Detailed Description	586
7.10.2	Function Documentation	587
7.10.2.1	nppiFilter_8u_C1R	587
7.10.2.2	nppiFilter_8u_C4R	587
7.10.2.3	nppiFilterBox_8u_C1R	588
7.10.2.4	nppiFilterBox_8u_C4R	588
7.10.2.5	nppiFilterColumn_8u_C1R	589
7.10.2.6	nppiFilterColumn_8u_C4R	589
7.10.2.7	nppiFilterMax_8u_C1R	590
7.10.2.8	nppiFilterMax_8u_C4R	590
7.10.2.9	nppiFilterMin_8u_C1R	591
7.10.2.10	nppiFilterMin_8u_C4R	591
7.10.2.11	nppiFilterRow_8u_C1R	591
7.10.2.12	nppiFilterRow_8u_C4R	592
7.10.2.13	nppiSumWindowColumn_8u32f_C1R	593
7.10.2.14	nppiSumWindowRow_8u32f_C1R	593
7.11	Morphological Operations	594

7.11.1	Detailed Description	594
7.11.2	Function Documentation	594
7.11.2.1	nppiDilate_8u_C1R	594
7.11.2.2	nppiDilate_8u_C4R	595
7.11.2.3	nppiErode_8u_C1R	595
7.11.2.4	nppiErode_8u_C4R	596
7.12	Image Linear Transforms	597
7.12.1	Detailed Description	597
7.12.2	Function Documentation	597
7.12.2.1	nppiMagnitude_32fc32f_C1R	597
7.12.2.2	nppiMagnitudeSqr_32fc32f_C1R	597
7.13	Compression	599
7.13.1	Detailed Description	599
7.13.2	Function Documentation	599
7.13.2.1	nppiDCTQuantFwd8x8LS_JPEG_8u16s_C1R	599
7.13.2.2	nppiDCTQuantInv8x8LS_JPEG_16s8u_C1R	600
7.13.2.3	nppiQuantFwdRawTableInit_JPEG_8u	600
7.13.2.4	nppiQuantFwdTableInit_JPEG_8u16u	601
7.13.2.5	nppiQuantInvTableInit_JPEG_8u16u	601
7.14	Geometric Transforms	602
7.14.1	Detailed Description	620
7.14.2	Function Documentation	620
7.14.2.1	nppiGetAffineBound	620
7.14.2.2	nppiGetAffineQuad	621
7.14.2.3	nppiGetAffineTransform	621
7.14.2.4	nppiGetPerspectiveBound	621
7.14.2.5	nppiGetPerspectiveQuad	622
7.14.2.6	nppiGetPerspectiveTransform	622
7.14.2.7	nppiGetRotateBound	623
7.14.2.8	nppiGetRotateQuad	623
7.14.2.9	nppiMirror_16u_AC4R	623
7.14.2.10	nppiMirror_16u_C1R	624
7.14.2.11	nppiMirror_16u_C3R	624
7.14.2.12	nppiMirror_16u_C4R	625
7.14.2.13	nppiMirror_32f_AC4R	625
7.14.2.14	nppiMirror_32f_C1R	625

7.14.2.15 nppiMirror_32f_C3R	626
7.14.2.16 nppiMirror_32f_C4R	626
7.14.2.17 nppiMirror_32s_AC4R	627
7.14.2.18 nppiMirror_32s_C1R	627
7.14.2.19 nppiMirror_32s_C3R	627
7.14.2.20 nppiMirror_32s_C4R	628
7.14.2.21 nppiMirror_8u_AC4R	628
7.14.2.22 nppiMirror_8u_C1R	629
7.14.2.23 nppiMirror_8u_C3R	629
7.14.2.24 nppiMirror_8u_C4R	629
7.14.2.25 nppiResize_8u_C1R	630
7.14.2.26 nppiResize_8u_C4R	630
7.14.2.27 nppiRotate_16u_AC4R	631
7.14.2.28 nppiRotate_16u_C1R	632
7.14.2.29 nppiRotate_16u_C3R	632
7.14.2.30 nppiRotate_16u_C4R	633
7.14.2.31 nppiRotate_32f_AC4R	633
7.14.2.32 nppiRotate_32f_C1R	634
7.14.2.33 nppiRotate_32f_C3R	634
7.14.2.34 nppiRotate_32f_C4R	635
7.14.2.35 nppiRotate_8u_AC4R	635
7.14.2.36 nppiRotate_8u_C1R	636
7.14.2.37 nppiRotate_8u_C3R	636
7.14.2.38 nppiRotate_8u_C4R	637
7.14.2.39 nppiWarpAffine_16u_AC4R	637
7.14.2.40 nppiWarpAffine_16u_C1R	637
7.14.2.41 nppiWarpAffine_16u_C3R	638
7.14.2.42 nppiWarpAffine_16u_C4R	639
7.14.2.43 nppiWarpAffine_16u_P3R	639
7.14.2.44 nppiWarpAffine_16u_P4R	639
7.14.2.45 nppiWarpAffine_32f_AC4R	639
7.14.2.46 nppiWarpAffine_32f_C1R	639
7.14.2.47 nppiWarpAffine_32f_C3R	640
7.14.2.48 nppiWarpAffine_32f_C4R	641
7.14.2.49 nppiWarpAffine_32f_P3R	641
7.14.2.50 nppiWarpAffine_32f_P4R	641

7.14.2.51 nppiWarpAffine_32s_AC4R	641
7.14.2.52 nppiWarpAffine_32s_C1R	641
7.14.2.53 nppiWarpAffine_32s_C3R	642
7.14.2.54 nppiWarpAffine_32s_C4R	643
7.14.2.55 nppiWarpAffine_32s_P3R	643
7.14.2.56 nppiWarpAffine_32s_P4R	643
7.14.2.57 nppiWarpAffine_8u_AC4R	643
7.14.2.58 nppiWarpAffine_8u_C1R	643
7.14.2.59 nppiWarpAffine_8u_C3R	645
7.14.2.60 nppiWarpAffine_8u_C4R	645
7.14.2.61 nppiWarpAffine_8u_P3R	645
7.14.2.62 nppiWarpAffine_8u_P4R	645
7.14.2.63 nppiWarpAffineBack_16u_AC4R	645
7.14.2.64 nppiWarpAffineBack_16u_C1R	646
7.14.2.65 nppiWarpAffineBack_16u_C3R	647
7.14.2.66 nppiWarpAffineBack_16u_C4R	647
7.14.2.67 nppiWarpAffineBack_16u_P3R	647
7.14.2.68 nppiWarpAffineBack_16u_P4R	647
7.14.2.69 nppiWarpAffineBack_32f_AC4R	648
7.14.2.70 nppiWarpAffineBack_32f_C1R	648
7.14.2.71 nppiWarpAffineBack_32f_C3R	649
7.14.2.72 nppiWarpAffineBack_32f_C4R	649
7.14.2.73 nppiWarpAffineBack_32f_P3R	649
7.14.2.74 nppiWarpAffineBack_32f_P4R	649
7.14.2.75 nppiWarpAffineBack_32s_AC4R	650
7.14.2.76 nppiWarpAffineBack_32s_C1R	650
7.14.2.77 nppiWarpAffineBack_32s_C3R	651
7.14.2.78 nppiWarpAffineBack_32s_C4R	651
7.14.2.79 nppiWarpAffineBack_32s_P3R	651
7.14.2.80 nppiWarpAffineBack_32s_P4R	651
7.14.2.81 nppiWarpAffineBack_8u_AC4R	652
7.14.2.82 nppiWarpAffineBack_8u_C1R	652
7.14.2.83 nppiWarpAffineBack_8u_C3R	653
7.14.2.84 nppiWarpAffineBack_8u_C4R	653
7.14.2.85 nppiWarpAffineBack_8u_P3R	653
7.14.2.86 nppiWarpAffineBack_8u_P4R	654

7.14.2.87 nppiWarpAffineQuad_16u_AC4R	654
7.14.2.88 nppiWarpAffineQuad_16u_C1R	654
7.14.2.89 nppiWarpAffineQuad_16u_C3R	655
7.14.2.90 nppiWarpAffineQuad_16u_C4R	655
7.14.2.91 nppiWarpAffineQuad_16u_P3R	655
7.14.2.92 nppiWarpAffineQuad_16u_P4R	656
7.14.2.93 nppiWarpAffineQuad_32f_AC4R	656
7.14.2.94 nppiWarpAffineQuad_32f_C1R	656
7.14.2.95 nppiWarpAffineQuad_32f_C3R	657
7.14.2.96 nppiWarpAffineQuad_32f_C4R	657
7.14.2.97 nppiWarpAffineQuad_32f_P3R	657
7.14.2.98 nppiWarpAffineQuad_32f_P4R	657
7.14.2.99 nppiWarpAffineQuad_32s_AC4R	658
7.14.2.100 nppiWarpAffineQuad_32s_C1R	658
7.14.2.101 nppiWarpAffineQuad_32s_C3R	659
7.14.2.102 nppiWarpAffineQuad_32s_C4R	659
7.14.2.103 nppiWarpAffineQuad_32s_P3R	659
7.14.2.104 nppiWarpAffineQuad_32s_P4R	659
7.14.2.105 nppiWarpAffineQuad_8u_AC4R	659
7.14.2.106 nppiWarpAffineQuad_8u_C1R	660
7.14.2.107 nppiWarpAffineQuad_8u_C3R	661
7.14.2.108 nppiWarpAffineQuad_8u_C4R	661
7.14.2.109 nppiWarpAffineQuad_8u_P3R	661
7.14.2.110 nppiWarpAffineQuad_8u_P4R	661
7.14.2.111 nppiWarpPerspective_16u_AC4R	661
7.14.2.112 nppiWarpPerspective_16u_C1R	662
7.14.2.113 nppiWarpPerspective_16u_C3R	663
7.14.2.114 nppiWarpPerspective_16u_C4R	663
7.14.2.115 nppiWarpPerspective_16u_P3R	663
7.14.2.116 nppiWarpPerspective_16u_P4R	663
7.14.2.117 nppiWarpPerspective_32f_AC4R	664
7.14.2.118 nppiWarpPerspective_32f_C1R	664
7.14.2.119 nppiWarpPerspective_32f_C3R	665
7.14.2.120 nppiWarpPerspective_32f_C4R	665
7.14.2.121 nppiWarpPerspective_32f_P3R	665
7.14.2.122 nppiWarpPerspective_32f_P4R	665

7.14.2.123nppiWarpPerspective_32s_AC4R	666
7.14.2.124nppiWarpPerspective_32s_C1R	666
7.14.2.125nppiWarpPerspective_32s_C3R	667
7.14.2.126nppiWarpPerspective_32s_C4R	667
7.14.2.127nppiWarpPerspective_32s_P3R	667
7.14.2.128nppiWarpPerspective_32s_P4R	667
7.14.2.129nppiWarpPerspective_8u_AC4R	668
7.14.2.130nppiWarpPerspective_8u_C1R	668
7.14.2.131nppiWarpPerspective_8u_C3R	669
7.14.2.132nppiWarpPerspective_8u_C4R	669
7.14.2.133nppiWarpPerspective_8u_P3R	669
7.14.2.134nppiWarpPerspective_8u_P4R	670
7.14.2.135nppiWarpPerspectiveBack_16u_AC4R	670
7.14.2.136nppiWarpPerspectiveBack_16u_C1R	670
7.14.2.137nppiWarpPerspectiveBack_16u_C3R	671
7.14.2.138nppiWarpPerspectiveBack_16u_C4R	671
7.14.2.139nppiWarpPerspectiveBack_16u_P3R	672
7.14.2.140nppiWarpPerspectiveBack_16u_P4R	672
7.14.2.141nppiWarpPerspectiveBack_32f_AC4R	672
7.14.2.142nppiWarpPerspectiveBack_32f_C1R	672
7.14.2.143nppiWarpPerspectiveBack_32f_C3R	673
7.14.2.144nppiWarpPerspectiveBack_32f_C4R	673
7.14.2.145nppiWarpPerspectiveBack_32f_P3R	674
7.14.2.146nppiWarpPerspectiveBack_32f_P4R	674
7.14.2.147nppiWarpPerspectiveBack_32s_AC4R	674
7.14.2.148nppiWarpPerspectiveBack_32s_C1R	674
7.14.2.149nppiWarpPerspectiveBack_32s_C3R	675
7.14.2.150nppiWarpPerspectiveBack_32s_C4R	675
7.14.2.151nppiWarpPerspectiveBack_32s_P3R	676
7.14.2.152nppiWarpPerspectiveBack_32s_P4R	676
7.14.2.153nppiWarpPerspectiveBack_8u_AC4R	676
7.14.2.154nppiWarpPerspectiveBack_8u_C1R	676
7.14.2.155nppiWarpPerspectiveBack_8u_C3R	677
7.14.2.156nppiWarpPerspectiveBack_8u_C4R	678
7.14.2.157nppiWarpPerspectiveBack_8u_P3R	678
7.14.2.158nppiWarpPerspectiveBack_8u_P4R	678

7.14.2.159	nppiWarpPerspectiveQuad_16u_AC4R	678
7.14.2.160	nppiWarpPerspectiveQuad_16u_C1R	678
7.14.2.161	nppiWarpPerspectiveQuad_16u_C3R	679
7.14.2.162	nppiWarpPerspectiveQuad_16u_C4R	680
7.14.2.163	nppiWarpPerspectiveQuad_16u_P3R	680
7.14.2.164	nppiWarpPerspectiveQuad_16u_P4R	680
7.14.2.165	nppiWarpPerspectiveQuad_32f_AC4R	680
7.14.2.166	nppiWarpPerspectiveQuad_32f_C1R	680
7.14.2.167	nppiWarpPerspectiveQuad_32f_C3R	681
7.14.2.168	nppiWarpPerspectiveQuad_32f_C4R	682
7.14.2.169	nppiWarpPerspectiveQuad_32f_P3R	682
7.14.2.170	nppiWarpPerspectiveQuad_32f_P4R	682
7.14.2.171	nppiWarpPerspectiveQuad_32s_AC4R	682
7.14.2.172	nppiWarpPerspectiveQuad_32s_C1R	682
7.14.2.173	nppiWarpPerspectiveQuad_32s_C3R	683
7.14.2.174	nppiWarpPerspectiveQuad_32s_C4R	683
7.14.2.175	nppiWarpPerspectiveQuad_32s_P3R	684
7.14.2.176	nppiWarpPerspectiveQuad_32s_P4R	684
7.14.2.177	nppiWarpPerspectiveQuad_8u_AC4R	684
7.14.2.178	nppiWarpPerspectiveQuad_8u_C1R	684
7.14.2.179	nppiWarpPerspectiveQuad_8u_C3R	685
7.14.2.180	nppiWarpPerspectiveQuad_8u_C4R	685
7.14.2.181	nppiWarpPerspectiveQuad_8u_P3R	686
7.14.2.182	nppiWarpPerspectiveQuad_8u_P4R	686
7.15	Color Conversion	687
7.15.1	Detailed Description	689
7.15.2	Function Documentation	689
7.15.2.1	nppiColorTwist32f_8u_AC4R	689
7.15.2.2	nppiColorTwist32f_8u_C3R	689
7.15.2.3	nppiColorTwist32f_8u_P3R	690
7.15.2.4	nppiLUT_Linear_8u_AC4R	690
7.15.2.5	nppiLUT_Linear_8u_C1R	691
7.15.2.6	nppiLUT_Linear_8u_C3R	691
7.15.2.7	nppiRGBToYCbCr420_8u_C3P3R	692
7.15.2.8	nppiRGBToYCbCr422_8u_C3C2R	692
7.15.2.9	nppiRGBToYCbCr_8u_AC4R	692

7.15.2.10	nppiRGBToYCbCr_8u_C3R	693
7.15.2.11	nppiRGBToYCbCr_8u_P3R	693
7.15.2.12	nppiYCbCr420ToRGB_8u_P3C3R	693
7.15.2.13	nppiYCbCr420ToYCbCr411_8u_P3P2R	694
7.15.2.14	nppiYCbCr420ToYCbCr422_8u_P3R	694
7.15.2.15	nppiYCbCr422ToRGB_8u_C2C3R	694
7.15.2.16	nppiYCbCr422ToYCbCr411_8u_P3R	695
7.15.2.17	nppiYCbCr422ToYCbCr420_8u_P3R	695
7.15.2.18	nppiYCbCrToRGB_8u_AC4R	695
7.15.2.19	nppiYCbCrToRGB_8u_C3R	696
7.15.2.20	nppiYCbCrToRGB_8u_P3R	696
7.16	Labeling and Segmentation	697
7.16.1	Detailed Description	697
7.16.2	Typedef Documentation	698
7.16.2.1	NppiGraphcutState	698
7.16.3	Function Documentation	698
7.16.3.1	nppiGraphcut8_32s8u	698
7.16.3.2	nppiGraphcut8GetSize	699
7.16.3.3	nppiGraphcut8InitAlloc	699
7.16.3.4	nppiGraphcut_32s8u	699
7.16.3.5	nppiGraphcutFree	700
7.16.3.6	nppiGraphcutGetSize	701
7.16.3.7	nppiGraphcutInitAlloc	701
7.17	NPP Signal Processing	702
7.17.1	Function Documentation	740
7.17.1.1	npps10Log10_32s_ISfs	740
7.17.1.2	npps10Log10_32s_Sfs	740
7.17.1.3	nppsAbs_16s	740
7.17.1.4	nppsAbs_16s_I	741
7.17.1.5	nppsAbs_32f	741
7.17.1.6	nppsAbs_32f_I	741
7.17.1.7	nppsAbs_32s	741
7.17.1.8	nppsAbs_32s_I	742
7.17.1.9	nppsAbs_64f	742
7.17.1.10	nppsAbs_64f_I	742
7.17.1.11	nppsAdd_16s	742

7.17.1.12 nppsAdd_16s32f	743
7.17.1.13 nppsAdd_16s32s_I	743
7.17.1.14 nppsAdd_16s_I	743
7.17.1.15 nppsAdd_16s_ISfs	744
7.17.1.16 nppsAdd_16s_Sfs	744
7.17.1.17 nppsAdd_16sc_ISfs	744
7.17.1.18 nppsAdd_16sc_Sfs	745
7.17.1.19 nppsAdd_16u	745
7.17.1.20 nppsAdd_16u_ISfs	745
7.17.1.21 nppsAdd_16u_Sfs	746
7.17.1.22 nppsAdd_32f	746
7.17.1.23 nppsAdd_32f_I	746
7.17.1.24 nppsAdd_32fc	747
7.17.1.25 nppsAdd_32fc_I	747
7.17.1.26 nppsAdd_32s_ISfs	747
7.17.1.27 nppsAdd_32s_Sfs	748
7.17.1.28 nppsAdd_32sc_ISfs	748
7.17.1.29 nppsAdd_32sc_Sfs	748
7.17.1.30 nppsAdd_32u	749
7.17.1.31 nppsAdd_64f	749
7.17.1.32 nppsAdd_64f_I	749
7.17.1.33 nppsAdd_64fc	750
7.17.1.34 nppsAdd_64fc_I	750
7.17.1.35 nppsAdd_64s_Sfs	750
7.17.1.36 nppsAdd_8u16u	751
7.17.1.37 nppsAdd_8u_ISfs	751
7.17.1.38 nppsAdd_8u_Sfs	751
7.17.1.39 nppsAddC_16s_ISfs	752
7.17.1.40 nppsAddC_16s_Sfs	752
7.17.1.41 nppsAddC_16sc_ISfs	752
7.17.1.42 nppsAddC_16sc_Sfs	753
7.17.1.43 nppsAddC_16u_ISfs	753
7.17.1.44 nppsAddC_16u_Sfs	753
7.17.1.45 nppsAddC_32f	754
7.17.1.46 nppsAddC_32f_I	754
7.17.1.47 nppsAddC_32fc	754

7.17.1.48 nppsAddC_32fc_I	755
7.17.1.49 nppsAddC_32s_ISfs	755
7.17.1.50 nppsAddC_32s_Sfs	755
7.17.1.51 nppsAddC_32sc_ISfs	756
7.17.1.52 nppsAddC_32sc_Sfs	756
7.17.1.53 nppsAddC_64f	756
7.17.1.54 nppsAddC_64f_I	757
7.17.1.55 nppsAddC_64fc	757
7.17.1.56 nppsAddC_64fc_I	757
7.17.1.57 nppsAddC_8u_ISfs	757
7.17.1.58 nppsAddC_8u_Sfs	758
7.17.1.59 nppsAddProduct_16s32s_Sfs	758
7.17.1.60 nppsAddProduct_16s_Sfs	759
7.17.1.61 nppsAddProduct_32f	759
7.17.1.62 nppsAddProduct_32fc	759
7.17.1.63 nppsAddProduct_32s_Sfs	760
7.17.1.64 nppsAddProduct_64f	760
7.17.1.65 nppsAddProduct_64fc	760
7.17.1.66 nppsAddProductC_16s_ISfs	761
7.17.1.67 nppsAddProductC_16s_Sfs	761
7.17.1.68 nppsAddProductC_16sc_ISfs	761
7.17.1.69 nppsAddProductC_16sc_Sfs	762
7.17.1.70 nppsAddProductC_16u_ISfs	762
7.17.1.71 nppsAddProductC_16u_Sfs	762
7.17.1.72 nppsAddProductC_32f	763
7.17.1.73 nppsAddProductC_32f_I	763
7.17.1.74 nppsAddProductC_32fc	763
7.17.1.75 nppsAddProductC_32fc_I	764
7.17.1.76 nppsAddProductC_32s_ISfs	764
7.17.1.77 nppsAddProductC_32s_Sfs	764
7.17.1.78 nppsAddProductC_32sc_ISfs	765
7.17.1.79 nppsAddProductC_32sc_Sfs	765
7.17.1.80 nppsAddProductC_64f	765
7.17.1.81 nppsAddProductC_64f_I	766
7.17.1.82 nppsAddProductC_64fc	766
7.17.1.83 nppsAddProductC_64fc_I	766

7.17.1.84 nppsAddProductC_8u_ISfs	767
7.17.1.85 nppsAddProductC_8u_Sfs	767
7.17.1.86 nppsAnd_16u	767
7.17.1.87 nppsAnd_16u_I	768
7.17.1.88 nppsAnd_32u	768
7.17.1.89 nppsAnd_32u_I	768
7.17.1.90 nppsAnd_8u	768
7.17.1.91 nppsAnd_8u_I	769
7.17.1.92 nppsAndC_16u	769
7.17.1.93 nppsAndC_16u_I	769
7.17.1.94 nppsAndC_32u	770
7.17.1.95 nppsAndC_32u_I	770
7.17.1.96 nppsAndC_8u	770
7.17.1.97 nppsAndC_8u_I	771
7.17.1.98 nppsArctan_32f	771
7.17.1.99 nppsArctan_32f_I	771
7.17.1.100 nppsArctan_64f	771
7.17.1.101 nppsArctan_64f_I	772
7.17.1.102 nppsCauchy_32f_I	772
7.17.1.103 nppsCauchyD_32f_I	772
7.17.1.104 nppsCauchyDD2_32f_I	772
7.17.1.105 nppsCopy_16s	773
7.17.1.106 nppsCopy_16sc	773
7.17.1.107 nppsCopy_32f	773
7.17.1.108 nppsCopy_32fc	774
7.17.1.109 nppsCopy_32s	774
7.17.1.110 nppsCopy_32sc	774
7.17.1.111 nppsCopy_64fc	774
7.17.1.112 nppsCopy_64s	775
7.17.1.113 nppsCopy_64sc	775
7.17.1.114 nppsCopy_8u	775
7.17.1.115 nppsCubrt_32f	776
7.17.1.116 nppsCubrt_32s16s_Sfs	776
7.17.1.117 nppsDiv_16s_ISfs	776
7.17.1.118 nppsDiv_16s_Sfs	777
7.17.1.119 nppsDiv_16sc_ISfs	777

7.17.1.120nppsDiv_16sc_Sfs	777
7.17.1.121nppsDiv_16u_ISfs	778
7.17.1.122nppsDiv_16u_Sfs	778
7.17.1.123nppsDiv_32f	778
7.17.1.124nppsDiv_32f_I	779
7.17.1.125nppsDiv_32fc	779
7.17.1.126nppsDiv_32fc_I	779
7.17.1.127nppsDiv_32s16s_Sfs	779
7.17.1.128nppsDiv_32s_ISfs	780
7.17.1.129nppsDiv_32s_Sfs	780
7.17.1.130nppsDiv_64f	781
7.17.1.131nppsDiv_64f_I	781
7.17.1.132nppsDiv_64fc	781
7.17.1.133nppsDiv_64fc_I	782
7.17.1.134nppsDiv_8u_ISfs	782
7.17.1.135nppsDiv_8u_Sfs	782
7.17.1.136nppsDiv_Round_16s_ISfs	783
7.17.1.137nppsDiv_Round_16s_Sfs	783
7.17.1.138nppsDiv_Round_16u_ISfs	783
7.17.1.139nppsDiv_Round_16u_Sfs	784
7.17.1.140nppsDiv_Round_8u_ISfs	784
7.17.1.141nppsDiv_Round_8u_Sfs	784
7.17.1.142nppsDivC_16s_ISfs	785
7.17.1.143nppsDivC_16s_Sfs	785
7.17.1.144nppsDivC_16sc_ISfs	785
7.17.1.145nppsDivC_16sc_Sfs	786
7.17.1.146nppsDivC_16u_ISfs	786
7.17.1.147nppsDivC_16u_Sfs	786
7.17.1.148nppsDivC_32f	787
7.17.1.149nppsDivC_32f_I	787
7.17.1.150nppsDivC_32fc	787
7.17.1.151nppsDivC_32fc_I	788
7.17.1.152nppsDivC_32s_ISfs	788
7.17.1.153nppsDivC_32s_Sfs	788
7.17.1.154nppsDivC_32sc_ISfs	789
7.17.1.155nppsDivC_32sc_Sfs	789

7.17.1.156nppsDivC_64f	789
7.17.1.157nppsDivC_64f_I	790
7.17.1.158nppsDivC_64fc	790
7.17.1.159nppsDivC_64fc_I	790
7.17.1.160nppsDivC_8u_ISfs	790
7.17.1.161nppsDivC_8u_Sfs	791
7.17.1.162nppsDivCRev_16s	791
7.17.1.163nppsDivCRev_16s_I	792
7.17.1.164nppsDivCRev_16u	792
7.17.1.165nppsDivCRev_16u_I	792
7.17.1.166nppsDivCRev_32f	792
7.17.1.167nppsDivCRev_32f_I	793
7.17.1.168nppsDivCRev_32s	793
7.17.1.169nppsDivCRev_32s_I	793
7.17.1.170nppsDivCRev_64f	794
7.17.1.171nppsDivCRev_64f_I	794
7.17.1.172nppsDivCRev_8u	794
7.17.1.173nppsDivCRev_8u_I	795
7.17.1.174nppsExp_16s_ISfs	795
7.17.1.175nppsExp_16s_Sfs	795
7.17.1.176nppsExp_32f	795
7.17.1.177nppsExp_32f64f	796
7.17.1.178nppsExp_32f_I	796
7.17.1.179nppsExp_32s_ISfs	796
7.17.1.180nppsExp_32s_Sfs	797
7.17.1.181nppsExp_64f	797
7.17.1.182nppsExp_64f_I	797
7.17.1.183nppsExp_64s_ISfs	797
7.17.1.184nppsExp_64s_Sfs	798
7.17.1.185nppsFree	798
7.17.1.186nppsIntegral_32s	798
7.17.1.187nppsIntegralGetBufferSize_32s	798
7.17.1.188nppsLn_16s_ISfs	798
7.17.1.189nppsLn_16s_Sfs	799
7.17.1.190nppsLn_32f	799
7.17.1.191nppsLn_32f_I	799

7.17.1.192nppsLn_32s16s_Sfs	799
7.17.1.193nppsLn_32s_ISfs	800
7.17.1.194nppsLn_32s_Sfs	800
7.17.1.195nppsLn_64f	800
7.17.1.196nppsLn_64f32f	801
7.17.1.197nppsLn_64f_I	801
7.17.1.198nppsLShiftC_16s	801
7.17.1.199nppsLShiftC_16s_I	801
7.17.1.200nppsLShiftC_16u	802
7.17.1.201nppsLShiftC_16u_I	802
7.17.1.202nppsLShiftC_32s	802
7.17.1.203nppsLShiftC_32s_I	803
7.17.1.204nppsLShiftC_32u	803
7.17.1.205nppsLShiftC_32u_I	803
7.17.1.206nppsLShiftC_8u	803
7.17.1.207nppsLShiftC_8u_I	804
7.17.1.208nppsMalloc_16s	804
7.17.1.209nppsMalloc_16sc	804
7.17.1.210nppsMalloc_16u	805
7.17.1.211nppsMalloc_32f	805
7.17.1.212nppsMalloc_32fc	805
7.17.1.213nppsMalloc_32s	805
7.17.1.214nppsMalloc_32sc	806
7.17.1.215nppsMalloc_32u	806
7.17.1.216nppsMalloc_64f	806
7.17.1.217nppsMalloc_64fc	806
7.17.1.218nppsMalloc_64s	807
7.17.1.219nppsMalloc_64sc	807
7.17.1.220nppsMalloc_8u	807
7.17.1.221nppsMax_16s	807
7.17.1.222nppsMax_32f	808
7.17.1.223nppsMax_32s	808
7.17.1.224nppsMax_64f	808
7.17.1.225nppsMaxGetBufferSize_16s	809
7.17.1.226nppsMaxGetBufferSize_32f	809
7.17.1.227nppsMaxGetBufferSize_32s	809

7.17.1.228nppsMaxGetBufferSize_64f	809
7.17.1.229nppsMin_16s	810
7.17.1.230nppsMin_32f	810
7.17.1.231nppsMin_32s	810
7.17.1.232nppsMin_64f	811
7.17.1.233nppsMinGetBufferSize_16s	811
7.17.1.234nppsMinGetBufferSize_32f	811
7.17.1.235nppsMinGetBufferSize_32s	812
7.17.1.236nppsMinGetBufferSize_64f	812
7.17.1.237nppsMinMax_16s	812
7.17.1.238nppsMinMax_16u	813
7.17.1.239nppsMinMax_32f	813
7.17.1.240nppsMinMax_32s	813
7.17.1.241nppsMinMax_32u	814
7.17.1.242nppsMinMax_64f	814
7.17.1.243nppsMinMax_8u	814
7.17.1.244nppsMinMaxGetBufferSize_16s	815
7.17.1.245nppsMinMaxGetBufferSize_16u	815
7.17.1.246nppsMinMaxGetBufferSize_32f	815
7.17.1.247nppsMinMaxGetBufferSize_32s	815
7.17.1.248nppsMinMaxGetBufferSize_32u	816
7.17.1.249nppsMinMaxGetBufferSize_64f	816
7.17.1.250nppsMinMaxGetBufferSize_8u	816
7.17.1.251nppsMul_16s	816
7.17.1.252nppsMul_16s32f	817
7.17.1.253nppsMul_16s32s_Sfs	817
7.17.1.254nppsMul_16s_I	817
7.17.1.255nppsMul_16s_ISfs	818
7.17.1.256nppsMul_16s_Sfs	818
7.17.1.257nppsMul_16sc_ISfs	818
7.17.1.258nppsMul_16sc_Sfs	819
7.17.1.259nppsMul_16u16s_Sfs	819
7.17.1.260nppsMul_16u_ISfs	819
7.17.1.261nppsMul_16u_Sfs	820
7.17.1.262nppsMul_32f	820
7.17.1.263nppsMul_32f32fc	820

7.17.1.264nppsMul_32f32fc_I	821
7.17.1.265nppsMul_32f_I	821
7.17.1.266nppsMul_32fc	821
7.17.1.267nppsMul_32fc_I	822
7.17.1.268nppsMul_32s32sc_ISfs	822
7.17.1.269nppsMul_32s32sc_Sfs	822
7.17.1.270nppsMul_32s_ISfs	823
7.17.1.271nppsMul_32s_Sfs	823
7.17.1.272nppsMul_32sc_ISfs	823
7.17.1.273nppsMul_32sc_Sfs	824
7.17.1.274nppsMul_64f	824
7.17.1.275nppsMul_64f_I	824
7.17.1.276nppsMul_64fc	825
7.17.1.277nppsMul_64fc_I	825
7.17.1.278nppsMul_8u16u	825
7.17.1.279nppsMul_8u_ISfs	826
7.17.1.280nppsMul_8u_Sfs	826
7.17.1.281nppsMul_Low_32s_Sfs	826
7.17.1.282nppsMulC_16s_ISfs	827
7.17.1.283nppsMulC_16s_Sfs	827
7.17.1.284nppsMulC_16sc_ISfs	827
7.17.1.285nppsMulC_16sc_Sfs	828
7.17.1.286nppsMulC_16u_ISfs	828
7.17.1.287nppsMulC_16u_Sfs	828
7.17.1.288nppsMulC_32f	829
7.17.1.289nppsMulC_32f16s_Sfs	829
7.17.1.290nppsMulC_32f_I	829
7.17.1.291nppsMulC_32fc	830
7.17.1.292nppsMulC_32fc_I	830
7.17.1.293nppsMulC_32s_ISfs	830
7.17.1.294nppsMulC_32s_Sfs	831
7.17.1.295nppsMulC_32sc_ISfs	831
7.17.1.296nppsMulC_32sc_Sfs	831
7.17.1.297nppsMulC_64f	832
7.17.1.298nppsMulC_64f64s_ISfs	832
7.17.1.299nppsMulC_64f_I	832

7.17.1.300nppsMulC_64fc	833
7.17.1.301nppsMulC_64fc_I	833
7.17.1.302nppsMulC_8u_ISfs	833
7.17.1.303nppsMulC_8u_Sfs	834
7.17.1.304nppsMulC_Low_32f16s	834
7.17.1.305nppsNormalize_16s_Sfs	834
7.17.1.306nppsNormalize_16sc_Sfs	835
7.17.1.307nppsNormalize_32f	835
7.17.1.308nppsNormalize_32fc	835
7.17.1.309nppsNormalize_64f	836
7.17.1.310nppsNormalize_64fc	836
7.17.1.311nppsNot_16u	836
7.17.1.312nppsNot_16u_I	837
7.17.1.313nppsNot_32u	837
7.17.1.314nppsNot_32u_I	837
7.17.1.315nppsNot_8u	837
7.17.1.316nppsNot_8u_I	838
7.17.1.317nppsOr_16u	838
7.17.1.318nppsOr_16u_I	838
7.17.1.319nppsOr_32u	838
7.17.1.320nppsOr_32u_I	839
7.17.1.321nppsOr_8u	839
7.17.1.322nppsOr_8u_I	839
7.17.1.323nppsOrC_16u	840
7.17.1.324nppsOrC_16u_I	840
7.17.1.325nppsOrC_32u	840
7.17.1.326nppsOrC_32u_I	841
7.17.1.327nppsOrC_8u	841
7.17.1.328nppsOrC_8u_I	841
7.17.1.329nppsRShiftC_16s	841
7.17.1.330nppsRShiftC_16s_I	842
7.17.1.331nppsRShiftC_16u	842
7.17.1.332nppsRShiftC_16u_I	842
7.17.1.333nppsRShiftC_32s	843
7.17.1.334nppsRShiftC_32s_I	843
7.17.1.335nppsRShiftC_32u	843

7.17.1.336nppsRShiftC_32u_I	844
7.17.1.337nppsRShiftC_8u	844
7.17.1.338nppsRShiftC_8u_I	844
7.17.1.339nppsSet_16s	844
7.17.1.340nppsSet_16sc	845
7.17.1.341nppsSet_32f	845
7.17.1.342nppsSet_32fc	845
7.17.1.343nppsSet_32s	846
7.17.1.344nppsSet_32sc	846
7.17.1.345nppsSet_64f	846
7.17.1.346nppsSet_64fc	846
7.17.1.347nppsSet_64s	847
7.17.1.348nppsSet_64sc	847
7.17.1.349nppsSet_8u	847
7.17.1.350nppsSqr_16s_ISfs	848
7.17.1.351nppsSqr_16s_Sfs	848
7.17.1.352nppsSqr_16sc_ISfs	848
7.17.1.353nppsSqr_16sc_Sfs	848
7.17.1.354nppsSqr_16u_ISfs	849
7.17.1.355nppsSqr_16u_Sfs	849
7.17.1.356nppsSqr_32f	849
7.17.1.357nppsSqr_32f_I	850
7.17.1.358nppsSqr_32fc	850
7.17.1.359nppsSqr_32fc_I	850
7.17.1.360nppsSqr_64f	850
7.17.1.361nppsSqr_64f_I	851
7.17.1.362nppsSqr_64fc	851
7.17.1.363nppsSqr_64fc_I	851
7.17.1.364nppsSqr_8u_ISfs	851
7.17.1.365nppsSqr_8u_Sfs	852
7.17.1.366nppsSqr_16s_ISfs	852
7.17.1.367nppsSqr_16s_Sfs	852
7.17.1.368nppsSqr_16sc_ISfs	853
7.17.1.369nppsSqr_16sc_Sfs	853
7.17.1.370nppsSqr_16u_ISfs	853
7.17.1.371nppsSqr_16u_Sfs	853

7.17.1.372nppsSqrt_32f	854
7.17.1.373nppsSqrt_32f_I	854
7.17.1.374nppsSqrt_32fc	854
7.17.1.375nppsSqrt_32fc_I	855
7.17.1.376nppsSqrt_32s16s_Sfs	855
7.17.1.377nppsSqrt_64f	855
7.17.1.378nppsSqrt_64f_I	855
7.17.1.379nppsSqrt_64fc	856
7.17.1.380nppsSqrt_64fc_I	856
7.17.1.381nppsSqrt_64s16s_Sfs	856
7.17.1.382nppsSqrt_64s_ISfs	856
7.17.1.383nppsSqrt_64s_Sfs	857
7.17.1.384nppsSqrt_8u_ISfs	857
7.17.1.385nppsSqrt_8u_Sfs	857
7.17.1.386nppsSub_16s	858
7.17.1.387nppsSub_16s32f	858
7.17.1.388nppsSub_16s_I	858
7.17.1.389nppsSub_16s_ISfs	859
7.17.1.390nppsSub_16s_Sfs	859
7.17.1.391nppsSub_16sc_ISfs	859
7.17.1.392nppsSub_16sc_Sfs	860
7.17.1.393nppsSub_16u_ISfs	860
7.17.1.394nppsSub_16u_Sfs	860
7.17.1.395nppsSub_32f	861
7.17.1.396nppsSub_32f_I	861
7.17.1.397nppsSub_32fc	861
7.17.1.398nppsSub_32fc_I	862
7.17.1.399nppsSub_32s_ISfs	862
7.17.1.400nppsSub_32s_Sfs	862
7.17.1.401nppsSub_32sc_ISfs	863
7.17.1.402nppsSub_32sc_Sfs	863
7.17.1.403nppsSub_64f	863
7.17.1.404nppsSub_64f_I	864
7.17.1.405nppsSub_64fc	864
7.17.1.406nppsSub_64fc_I	864
7.17.1.407nppsSub_8u_ISfs	864

7.17.1.408nppsSub_8u_Sfs	865
7.17.1.409nppsSubC_16s_ISfs	865
7.17.1.410nppsSubC_16s_Sfs	866
7.17.1.411nppsSubC_16sc_ISfs	866
7.17.1.412nppsSubC_16sc_Sfs	866
7.17.1.413nppsSubC_16u_ISfs	867
7.17.1.414nppsSubC_16u_Sfs	867
7.17.1.415nppsSubC_32f	867
7.17.1.416nppsSubC_32f_I	868
7.17.1.417nppsSubC_32fc	868
7.17.1.418nppsSubC_32fc_I	868
7.17.1.419nppsSubC_32s_ISfs	868
7.17.1.420nppsSubC_32s_Sfs	869
7.17.1.421nppsSubC_32sc_ISfs	869
7.17.1.422nppsSubC_32sc_Sfs	870
7.17.1.423nppsSubC_64f	870
7.17.1.424nppsSubC_64f_I	870
7.17.1.425nppsSubC_64fc	871
7.17.1.426nppsSubC_64fc_I	871
7.17.1.427nppsSubC_8u_ISfs	871
7.17.1.428nppsSubC_8u_Sfs	872
7.17.1.429nppsSubCRev_16s_ISfs	872
7.17.1.430nppsSubCRev_16s_Sfs	872
7.17.1.431nppsSubCRev_16sc_ISfs	873
7.17.1.432nppsSubCRev_16sc_Sfs	873
7.17.1.433nppsSubCRev_16u_ISfs	873
7.17.1.434nppsSubCRev_16u_Sfs	874
7.17.1.435nppsSubCRev_32f	874
7.17.1.436nppsSubCRev_32f_I	874
7.17.1.437nppsSubCRev_32fc	875
7.17.1.438nppsSubCRev_32fc_I	875
7.17.1.439nppsSubCRev_32s_ISfs	875
7.17.1.440nppsSubCRev_32s_Sfs	876
7.17.1.441nppsSubCRev_32sc_ISfs	876
7.17.1.442nppsSubCRev_32sc_Sfs	876
7.17.1.443nppsSubCRev_64f	877

7.17.1.444	nppsSubCRev_64f_I	877
7.17.1.445	nppsSubCRev_64fc	877
7.17.1.446	nppsSubCRev_64fc_I	878
7.17.1.447	nppsSubCRev_8u_ISfs	878
7.17.1.448	nppsSubCRev_8u_Sfs	878
7.17.1.449	nppsSum_16s32s_Sfs	879
7.17.1.450	nppsSum_16s_Sfs	879
7.17.1.451	nppsSum_16sc32sc_Sfs	879
7.17.1.452	nppsSum_16sc_Sfs	880
7.17.1.453	nppsSum_32f	880
7.17.1.454	nppsSum_32fc	880
7.17.1.455	nppsSum_32s_Sfs	881
7.17.1.456	nppsSum_64f	881
7.17.1.457	nppsSum_64fc	881
7.17.1.458	nppsSumGetBufferSize_16s32s_Sfs	882
7.17.1.459	nppsSumGetBufferSize_16s_Sfs	882
7.17.1.460	nppsSumGetBufferSize_16sc32sc_Sfs	882
7.17.1.461	nppsSumGetBufferSize_16sc_Sfs	882
7.17.1.462	nppsSumGetBufferSize_32f	883
7.17.1.463	nppsSumGetBufferSize_32fc	883
7.17.1.464	nppsSumGetBufferSize_32s_Sfs	883
7.17.1.465	nppsSumGetBufferSize_64f	884
7.17.1.466	nppsSumGetBufferSize_64fc	884
7.17.1.467	nppsXor_16u	884
7.17.1.468	nppsXor_16u_I	885
7.17.1.469	nppsXor_32u	885
7.17.1.470	nppsXor_32u_I	885
7.17.1.471	nppsXor_8u	885
7.17.1.472	nppsXor_8u_I	886
7.17.1.473	nppsXorC_16u	886
7.17.1.474	nppsXorC_16u_I	886
7.17.1.475	nppsXorC_32u	887
7.17.1.476	nppsXorC_32u_I	887
7.17.1.477	nppsXorC_8u	887
7.17.1.478	nppsXorC_8u_I	888
7.17.1.479	nppsZero_16s	888

7.17.1.480	nppsZero_16sc	888
7.17.1.481	nppsZero_32f	888
7.17.1.482	nppsZero_32fc	889
7.17.1.483	nppsZero_32s	889
7.17.1.484	nppsZero_32sc	889
7.17.1.485	nppsZero_64f	889
7.17.1.486	nppsZero_64fc	890
7.17.1.487	nppsZero_64s	890
7.17.1.488	nppsZero_64sc	890
7.17.1.489	nppsZero_8u	890
8	Data Structure Documentation	891
8.1	Npp16sc Struct Reference	891
8.1.1	Detailed Description	891
8.1.2	Field Documentation	891
8.1.2.1	im	891
8.1.2.2	re	891
8.2	Npp32fc Struct Reference	892
8.2.1	Detailed Description	892
8.2.2	Field Documentation	892
8.2.2.1	im	892
8.2.2.2	re	892
8.3	Npp32sc Struct Reference	893
8.3.1	Detailed Description	893
8.3.2	Field Documentation	893
8.3.2.1	im	893
8.3.2.2	re	893
8.4	Npp64fc Struct Reference	894
8.4.1	Detailed Description	894
8.4.2	Field Documentation	894
8.4.2.1	im	894
8.4.2.2	re	894
8.5	Npp64sc Struct Reference	895
8.5.1	Detailed Description	895
8.5.2	Field Documentation	895
8.5.2.1	im	895
8.5.2.2	re	895

8.6	NppiHaarBuffer Struct Reference	896
8.6.1	Field Documentation	896
8.6.1.1	haarBuffer	896
8.6.1.2	haarBufferSize	896
8.7	NppiHaarClassifier_32f Struct Reference	897
8.7.1	Field Documentation	897
8.7.1.1	classifiers	897
8.7.1.2	classifierSize	897
8.7.1.3	classifierStep	897
8.7.1.4	counterDevice	897
8.7.1.5	numClassifiers	897
8.8	NppiPoint Struct Reference	898
8.8.1	Detailed Description	898
8.8.2	Field Documentation	898
8.8.2.1	x	898
8.8.2.2	y	898
8.9	NppiRect Struct Reference	899
8.9.1	Detailed Description	899
8.9.2	Field Documentation	899
8.9.2.1	height	899
8.9.2.2	width	899
8.9.2.3	x	899
8.9.2.4	y	899
8.10	NppiSize Struct Reference	900
8.10.1	Detailed Description	900
8.10.2	Field Documentation	900
8.10.2.1	height	900
8.10.2.2	width	900
8.11	NppLibraryVersion Struct Reference	901
8.11.1	Field Documentation	901
8.11.1.1	build	901
8.11.1.2	major	901
8.11.1.3	minor	901

Chapter 1

NVIDIA Performance Primitives

1.1 What is NPP?

NVIDIA NPP is a library of functions for performing CUDA accelerated processing. The initial set of functionality in the library focuses on imaging and video processing and is widely applicable for developers in these areas. NPP will evolve over time to encompass more of the compute heavy tasks in a variety of problem domains. The NPP library is written to maximize flexibility, while maintaining high performance.

NPP can be used in one of two ways:

- A stand-alone library for adding GPU acceleration to an application with minimal effort. Using this route allows developers to add GPU acceleration to their applications in a matter of hours.
- A cooperative library for interoperating with a developer's GPU code efficiently.

Either route allows developers to harness the massive compute resources of NVIDIA GPUs, while simultaneously reducing development times.

1.2 Documentation

- [General API Conventions](#)
- [Signal-Processing Specific API Conventions](#)
- [Imaging-Processing Specific API Conventions](#)

1.3 Technical Specifications

Supported Platforms:

- Microsoft Windows 7 (64-bit and 32-bit)
- Microsoft Windows Vista (64-bit and 32-bit)
- Microsoft Windows XP (64-bit and 32-bit)
- Linux (Centos & Ubuntu) (64-bit and 32-bit)
- Mac OS X

1.4 Files

NPP is comprised of the following files:

1.4.1 Header Files

- [npp.h](#)
- [nppcore.h](#)
- [nppdefs.h](#)
- [nppi.h](#)
- [npps.h](#)
- [nppversion.h](#)

All those header files are located in the CUDA Toolkit's

`/include/`

directory.

1.4.2 Library Files

On the Windows platform the NPP stub library is found in the CUDA Toolkit's library directory:

`/lib/npp.lib`

The matching DLL is located in the CUDA Toolkit's binary directory:

```
/bin/npp32_32_7.dll    // Dynamic library for 32-bit Windows.  
/bin/npp64_32_7.dll    // Dynamic library for 64-bit Windows.
```

On Linux and Mac platforms the dynamic libraries are located in the lib directory

```
/lib/libnpp32.so.3.2.9  // NPP 32-bit dynamic library for Linux  
/lib/libnpp64.so.3.2.9  // NPP 64-bit dynamic library for Linux  
  
/lib/libnpp32.3.2.dylib // NPP 32-bit dynamic library for Mac  
/lib/libnpp64.3.2.dylib // NPP 64-bit dynamic library for Mac
```

1.5 Supported NVIDIA Hardware

NPP runs on all CUDA capable NVIDIA hardware. For details please see http://www.nvidia.com/object/cuda_learn_products.html

Chapter 2

General API Conventions

2.1 Memory Management

The design of all the NPP functions follows the same guidelines as other NVIDIA CUDA libraries like cuFFT and cuBLAS. That is that all pointer arguments in those APIs are device pointers.

This convention enables the individual developer to make smart choices about memory management that minimize the number of memory transfers. It also allows the user the maximum flexibility regarding which of the various memory transfer mechanisms offered by the CUDA runtime is used, e.g. synchronous or asynchronous memory transfers, zero-copy and pinned memory, etc.

The most basic steps involved in using NPP for processing data is as follows:

1. Transfer input data from the host to device using

```
cudaMemcpy(...)
```

2. Process data using one or several NPP functions or custom CUDA kernels
3. Transfer the result data from the device to the host using

```
cudaMemcpy(...)
```

2.1.1 Scratch Buffer and Host Pointer

Some primitives of NPP require additional device memory buffers (scratch buffers) for calculations, e.g. signal and image reductions (Sum, Max, Min, MinMax, etc.). In order to give the NPP user maximum control regarding memory allocations and performance, it is the user's responsibility to allocate and delete those temporary buffers. For one this has the benefit that the library will not allocate memory unbeknownst to the user. It also allows developers who invoke the same primitive repeatedly to allocate the scratch only once, improving performance and potential device-memory fragmentation.

Scratch-buffer memory is unstructured and may be passed to the primitive in uninitialized form. This allows for reuse of the same scratch buffers with any primitive require scratch memory, as long as it is sufficiently sized.

The minimum scratch-buffer size for a given primitive (e.g. [nppsSum_32f\(\)](#)) can be obtained by a companion function (e.g. [nppsSumGetBufferSize_32f\(\)](#)). The buffer size is returned via a host pointer as allocation of the scratch-buffer is performed via CUDA runtime host code.

An example to invoke Sum primitive and allocate and free the necessary scratch memory:

```
...
// Compute the appropriate size of the scratch-memory buffer
int nBufferSize;
nppsSumGetBufferSize_32f(nLength, &nBufferSize);
// Allocate the scratch buffer
Npp8u * pDeviceBuffer;
cudaMalloc((void **)&pDeviceBuffer, nBufferSize);
// Call the primitive with the scratch buffer
nppsSum_32f(pSrc, nLength, pSum, nppAlgHintNone, pDeviceBuffer);
// Free the scratch buffer
cudaFree(pDeviceBuffer);
...
```

2.2 Function Naming

Since NPP is a C API and therefore does not allow for function overloading for different data-types the NPP naming convention addresses the need to differentiate between different flavors of the same algorithm

or primitive function but for various data types. This disambiguation of different flavors of a primitive is done via a suffix containing data type and other disambiguating information.

In addition to the flavor suffix, all NPP functions are prefixed with by the letters "npp". Primitives belonging to NPP's image-processing module add the letter "i" to the npp prefix, i.e. are prefixed by "nppi". Similarly signal-processing primitives are prefixed with "npps".

The general naming scheme is:

`npp<module info><PrimitiveName>_<data-type info>[_<additional flavor info>](<parameter list>)`

The data-type information uses the same names as the [Basic NPP Data Types](#). For example the data-type information "8u" would imply that the primitive operates on [Npp8u](#) data.

If a primitive consumes different type data from what it produces, both types will be listed in the order of consumed to produced data type.

Details about the "additional flavor information" is provided for each of the NPP modules, since each problem domain uses different flavor information suffixes.

2.3 Integer Result Scaling

NPP signal processing and imaging primitives often operate on integer data. This integer data is a usually a fixed point fractional representation of some physical magnitue (e.g. luminance). Because of this fixed-point nature of the representation many numerical operations (e.g. addition or multiplication) tend produce results exceeding the original fixed-point range if treated as regular integers.

In cases where the results exceed the original range, these functions clamp the result values back to the valid range. E.g. the maximum positive value for a 16-bit unsigned integer is 32767. A multiplication operation of $4 * 10000 = 40000$ would exceed this range. The result would be clamped to be 32767.

To avoid the level of lost information due to clamping most integer primitives allow for result scaling. Primitives with result scaling have the "Sfs" suffix in their name and provide a parameter "nScaleFactor" that controls the amount of scaling. Before the results of an operation are clamped to the valid output-data range by multiplying them with $2^{-nScaleFactor}$.

Example: The primitive [nppsSqr_8u_Sfs\(\)](#) computes the square of 8-bit unsigned sample values in a signal (1D array of values). The maximum value of a 8-bit value is 255. The square of $255^2 = 65025$ which would be clamped to 255 if no result scaling is performed. In order to map the maximum value of 255 to 255 in the result, one would specify an integer result scaling factor of 8, i.e. multiply each result with $2^{-8} = \frac{1}{2^8} = \frac{1}{256}$. The final result for a signal value of 255 being squared and scaled would be:

$$255^2 \cdot 2^{-8} = 254.00390625$$

which would be rounded to a final result of 254.

A medium gray value of 128 would result in

$$128^2 \cdot 2^{-8} = 64$$

Chapter 3

Signal-Processing Specific API Conventions

3.1 Signal Data

Signal data is passed to and from NPPS primitives via a pointer to the signal's data type.

The general idea behind this fairly low-level way of passing signal data is ease-of-adoption into existing software projects:

- Passing the data pointer rather than a higher-level signal struct allows for easy adoption by not requiring a specific signal representation (that could include total signal size offset, or other additional information). This avoids awkward packing and unpacking of signal data from the host application to an NPP specific signal representation.

3.1.1 Parameter Names for Signal Data

There are three general cases of image-data passing throughout NPP detailed in the following sections.

Those are signals consumed by the algorithm.

3.1.1.1 Source Signal Pointer

The source signal data is generally passed via a pointer named

`pSrc`

The source signal pointer is generally defined constant, enforcing that the primitive does not change any image data pointed to by that pointer. E.g.

```
nppsPrimitive_32s(const Npp32s * pSrc, ...)
```

In case the primitive consumes multiple signals as inputs the source pointers are numbered like this:

`pSrc1, pSrc2, ...`

3.1.1.2 Destination Signal Pointer

The destination signal data is generally passed via a pointer named

`pDst`

In case the primitive consumes multiple signals as inputs the source pointers are numbered like this:

`pDst1, pDst2, ...`

3.1.1.3 In-Place Signal Pointer

In the case of in-place processing, source and destination are served by the same pointer and thus pointers to in-place signal data are called:

`pSrcDst`

3.1.2 Signal Data Alignment Requirements

NPP requires signal sample data to be naturally aligned, i.e. any pointer

```
NppType * p;
```

to a sample in a signal needs to fulfill:

```
assert(p % sizeof(p) == 0);
```

3.1.3 Signal Data Related Error Codes

All NPPI primitives operating on signal data validate the signal-data pointer for proper alignment and test that the point is not null.

Failed validation results in one of the following error codes being returned and the primitive not being executed:

- [NPP_NULL_POINTER_ERROR](#) is returned if the image-data pointer is 0 (NULL).
- [NPP_ALIGNMENT_ERROR](#) if the signal-data pointer address is not a multiple of the signal's data-type size.

3.2 Signal Length

The vast majority of NPPS functions take a

```
nLength
```

parameter that tells the primitive how many of the signal's samples starting from the given data pointer are to be processed.

3.2.1 Length Related Error Codes

All NPPS primitives taking a length parameter validate this input.

Failed validation results in the following error code being returned and the primitive not being executed:

- [NPP_SIZE_ERROR](#) is returned if the length is negative.

Chapter 4

Imaging-Processing Specific API Conventions

4.1 Function Naming

Image processing related functions use a number of suffixes to indicate various different flavors of a primitive beyond just different data types. The flavor suffix uses the following abbreviations:

- "A" if the image is a 4 channel image this indicates the result alpha channel is not affected by the primitive.
- "Cn" the image consists of n channel packed pixels, where n can be 1, 2, 3 or 4.
- "Pn" the image consists of n separate image planes, where n can be 1, 2, 3 or 4.
- "C" (following the channel information) indicates that the primitive only operates on one of the color channels, the "channel-of-interest". All other output channels are not affected by the primitive.
- "I" indicates that the primitive works "in-place". In this case the image-data pointer is usually named "pSrcDst" to indicate that the image data serves as source and destination at the same time.
- "M" indicates "masked operation". These types of primitives have an additional "mask image" as input. Each pixel in the destination image corresponds to a pixel in the mask image. Only pixels with a corresponding non-zero mask pixel are being processed.
- "R" indicates the primitive operates only on a rectangular "region-of-interest" or "ROI". All ROI primitives take an additional input parameter of type [NppiSize](#), which specifies the width and height of the rectangular region that the primitive should process. For details on how primitives operate on ROIs see: [Region-of-Interest \(ROI\)](#).
- "Sfs" indicates the result values are processed by fixed scaling and saturation before they're written out.

The suffixes above always appear in alphabetical order. E.g. a 4 channel primitive not affecting the alpha channel with masked operation, in place and with scaling/saturation and ROI would have the postfix: "AC4IMRSfs".

4.2 Image Data

Image data is passed to and from NPPI primitives via a pair of parameters:

1. A pointer to the image's underlying data type.
2. A line step in bytes (also sometimes called line stride).

The general idea behind this fairly low-level way of passing image data is ease-of-adoption into existing software projects:

- Passing a raw pointer to the underlying pixel data type, rather than structured (by color) channel pixel data allows usage of the function in a wide variety of situations avoiding risky type cast or expensive image data copies.
- Passing the data pointer and line step individually rather than a higher-level image struct again allows for easy adoption by not requiring a specific image representation and thus avoiding awkward packing and unpacking of image data from the host application to an NPP specific image representation.

4.2.1 Line Step

The line step (also called "line stride" or "row step") allows lines of oddly sized images to start on well-aligned addresses by adding a number of unused bytes at the ends of the lines. This type of line padding has been common practice in digital image processing for a long time and is not particular to GPU image processing.

The line step is the number of bytes in a line **including the padding**. An other way to interpret this number is to say that it is the number of bytes between the first pixel of successive rows in the image, or generally the number of bytes between two neighboring pixels in any column of pixels.

The general reason for the existence of the line step it is that uniformly aligned rows of pixel enable optimizations of memory-access patterns.

Even though all functions in NPP will work with arbitrarily aligned images, best performance can only be achieved with well aligned image data. Any image data allocated with the NPP image allocators or the 2D memory allocators in the CUDA runtime, is well aligned.

Particularly on older CUDA capable GPUs it is likely that the performance decrease for misaligned data is substantial (orders of magnitude).

All image data passed to NPPI primitives requires a line step to be provided. It is important to keep in mind that this line step is always specified in terms of bytes, not pixels.

4.2.2 Parameter Names for Image Data

There are three general cases of image-data passing throughout NPP detailed in the following sections.

4.2.2.1 Passing Source-Image Data

Those are images consumed by the algorithm.

4.2.2.1.1 Source-Image Pointer

The source image data is generally passed via a pointer named

```
pSrc
```

The source image pointer is generally defined constant, enforcing that the primitive does not change any image data pointed to by that pointer. E.g.

```
nppiPrimitive_32s_C1R(const Npp32s * pSrc, ...)
```

In case the primitive consumes multiple images as inputs the source pointers are numbered like this:

```
pSrc1, pSrc2, ...
```

4.2.2.1.2 Source-Image Line Step

The source-image line step is the number of bytes between successive rows in the image. The source-image line step parameter is

```
nSrcStep
```

or in the case of multiple source images

```
nSrcStep1, nSrcStep2, ...
```

4.2.2.2 Passing Destination-Image Data

Those are images produced by the algorithm.

4.2.2.2.1 Destination-Image Pointer

The destination image data is generally passed via a pointer named

```
pDst
```

In case the primitive consumes multiple images as inputs the source pointers are numbered like this:

```
pDst1, pDst2, ...
```

4.2.2.2.2 Destination-Image Line Step

The destination-image line step parameter is

```
nDstStep
```

or in the case of multiple destination images

```
nDstStep1, nDstStep2, ...
```

4.2.2.3 Passing In-Place Image Data

4.2.2.3.1 In-Place Image Pointer

In the case of in-place processing, source and destination are served by the same pointer and thus pointers to in-place image data are called:

```
pSrcDst
```

4.2.2.3.2 In-Place-Image Line Step

The in-place line step parameter is

```
nSrcDstStep
```

4.2.2.4 Passing Mask-Image Data

Some image processing primitives have variants supporting [Masked Operation](#).

4.2.2.4.1 Mask-Image Pointer

The mask-image data is generally passed via a pointer named

`pMask`

4.2.2.4.2 Mask-Image Line Step

The mask-image line step parameter is

`nMaskStep`

4.2.3 Image Data Alignment Requirements

NPP requires pixel data to adhere to certain alignment constraints: For 2 and 4 channel images the following alignment requirement holds: `data_pointer % (#channels * sizeof(channel type)) == 0`. E.g. a 4 channel image with underlying type `Npp8u` (8-bit unsigned) would require all pixels to fall on addresses that are multiples of 4 (4 channels * 1 byte size).

As a logical consequence of all pixels being aligned to their natural size the image line steps of 2 and 4 channel images also need to be multiples of the pixel size.

1 and 3 channel images only require that pixel pointers are aligned to the underlying data type, i.e. `pData % sizeof(data type) == 0`. And consequentially line steps are also held to this requirement.

4.2.4 Image Data Related Error Codes

All NPPI primitives operating on image data validate the image-data pointer for proper alignment and test that the point is not null. They also validate the line stride for proper alignment and guard against the step being less or equal to 0. Failed validation results in one of the following error codes being returned and the primitive not being executed:

- `NPP_STEP_ERROR` is returned if the data step is 0 or negative.
- `NPP_NOT_EVEN_STEP_ERROR` is returned if the line step is not a multiple of the pixel size for 2 and 4 channel images.
- `NPP_NULL_POINTER_ERROR` is returned if the image-data pointer is 0 (NULL).
- `NPP_ALIGNMENT_ERROR` if the image-data pointer address is not a multiple of the pixel size for 2 and 4 channel images.

4.3 Region-of-Interest (ROI)

In practice processing a rectangular sub-region of an image is often more common than processing complete images. The vast majority of NPP's image-processing primitives allow for processing of such sub regions also referred to as regions-of-interest or ROIs.

All primitives supporting ROI processing are marked by a "R" in their name suffix. Where possible, the ROI a primitive operates on is passed as a single `NppiSize` struct, which provides the with and height of the ROI. This raises the obvious question how the primitive knows where in the image this rectangle of (width, height) is located. The "start pixel" of the ROI is implicitly given by the image-data pointer. I.e. instead

of explicitly passing a pixel coordinate for the upper-right corner of the ROI the primitive's user needs to perform the necessary offset computation on the image data pointers, such that the pointers passed to the primitive thus point to the start of the ROI.

In practice this means that for an image (pSrc, nSrcStep) and the start-pixel of the ROI being given by (xROI, yROI), one would pass

```
pSrcOffset = pSrc + yROI * nSrcStep + xROI * PixelSize;
```

as the image-data source to the primitive. PixelSize is typically computed as

```
PixelSize = NumberOfColorChannels * sizeof(PixelDataType).
```

E.g. for a primitive like `nppiSet_16s_C4R()` we would have

- `NumberOfColorChannels == 4;`
- `sizeof(Npp16s) == 2;`
- and thus `PixelSize = 4 * 2 = 8;`

4.3.1 ROI Related Error Codes

All NPPI primitives operating on ROIs of image data validate the ROI size and image's step size. Failed validation results in one of the following error codes being returned and the primitive not being executed:

- `NPP_SIZE_ERROR` is returned if either the ROI width or ROI height are negative.
- `NPP_STEP_ERROR` is returned if the ROI width exceeds the image's line step. In mathematical terms $(\text{widthROI} * \text{PixelSize}) > \text{nLinStep}$ indicates an error.

4.4 Masked Operation

Some primitive support masked operation. An "M" in the suffix of those variants indicates masked operation. Primitives supporting masked operation consume an additional input image provided via a [Mask-Image Pointer](#) and [Mask-Image Line Step](#). The mask image is interpreted by these primitives as a boolean image. The values of type `Npp8u` are interpreted as boolean values where a values of 0 indicates false, any non-zero values true.

Unless otherwise indicated the operation is only performed on pixels where its spatially corresponding mask pixel is true (non-zero). E.g. a masked copy operation would only copy those pixels in the ROI that have corresponding non-zero mask pixels.

4.5 Channel-of-Interest API

Some primitives allow restricting operations to a single channel of interest within a multi-channel image. These primitives are suffixed with the letter "C" (after the channel information, e.g. `nppiCopy_8u_C3CR(...)`). The channel-of-interest is generally selected by offsetting the image-data pointer to point directly to the channel- of-interest rather than the base of the first pixel in the ROI.

4.5.1 Select-Channel Source-Image Pointer

This is a pointer to the channel-of-interest within the first pixel of the source image. E.g. if pSrc is the pointer to the first pixel inside the ROI of a three channel image. Using the appropriate select-channel copy

primitive one could copy the second channel of this source image into the first channel of a destination image given by pDst by offsetting the pointer by two data items:

```
nppiCopy_8u_C3CR(pSrc + 2, nSrcStep, pDst, nDstStep, oSizeROI);
```

4.5.2 Select-Channel Destination-Image Pointer

This is a pointer to the channel-of-interest within the first pixel of the destination image. E.g. if pDst is the pointer to the first pixel inside the ROI of a three channel image. Using the appropriate select-channel copy primitive one could copy data into the second channel of this destination image from the first channel of a source image given by pSrc by offsetting the destination pointer by two data items:

```
nppiCopy_8u_C3CR(pSrc, nSrcStep, pDst + 2, nDstStep, oSizeROI);
```

4.6 Geometric Transform API Specifics

This section covers some of the unique API features common to the geometric transform primitives.

4.6.1 Geometric Transforms and ROIs

Geometric transforms operate on source and destination ROIs. The way these ROIs affect the processing of pixels differs from other (non geometric) image-processing primitives: Only pixels in the intersection of the destination ROI and the transformed source ROI are being processed.

The typical processing proceeds as follows:

1. Transform the rectangular source ROI (given in source image coordinates) into the destination image space. This yields a quadrilateral.
2. Write only pixels in the intersection of the transformed source ROI and the destination ROI.

4.6.2 Pixel Interpolation

The majority of image geometry transform operation need to perform a resampling of the source image as source and destination pixels are not coincident.

NPP supports the following pixel interpolation modes (in order from fastest to slowest and lowest to highest quality):

- nearest neighbor
- linear interpolation
- cubic convolution
- supersampling
- interpolation using Lanczos window function

4.6.3 Rotate specific Error Codes

- [NPP_INTERPOLATION_ERROR](#) if eInterpolation has an illegal value.
- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1.
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) indicates an error condition if srcROIrect has no intersection with the source image.
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) indicates a warning that no operation is performed if the transformed source ROI does not intersect the destination ROI.

Chapter 5

Module Index

5.1 Modules

Here is a list of all modules:

NPP Core	23
NPP Type Definitions and Constants	26
Basic NPP Data Types	35
NPP Image Processing	37
Memory Management	39
Data-Exchange and Initialization	51
Arithmetic and Logical Operations	133
Threshold and Compare Operations	553
Statistics Functions	557
Filtering Functions	585
Morphological Operations	594
Image Linear Transforms	597
Compression	599
Geometric Transforms	602
Color Conversion	687
Labeling and Segmentation	697
NPP Signal Processing	702

Chapter 6

Data Structure Index

6.1 Data Structures

Here are the data structures with brief descriptions:

Npp16sc (Complex Number This struct represents a short complex number)	891
Npp32fc (Complex Number This struct represents a single floating-point complex number) . . .	892
Npp32sc (Complex Number This struct represents a signed int complex number)	893
Npp64fc (Complex Number This struct represents a double floating-point complex number) . .	894
Npp64sc (Complex Number This struct represents a long long complex number)	895
NppiHaarBuffer	896
NppiHaarClassifier_32f	897
NppiPoint (2D Point)	898
NppiRect (2D Rectangle This struct contains position and size information of a rectangle in two space)	899
NppiSize (2D Size This struct typically represents the size of a a rectangular region in two space)	900
NppLibraryVersion	901

Chapter 7

Module Documentation

7.1 NPP Core

Basic functions for library management, in particular library version and device property query functions.

Functions

- `const NppLibraryVersion * nppGetLibVersion (void)`
Get the NPP library version.
- `NppGpuComputeCapability nppGetGpuComputeCapability (void)`
What CUDA compute model is supported by the active CUDA device?
- `int nppGetGpuNumSMs (void)`
Get the number of Streaming Multiprocessors (SM) on the active CUDA device.
- `int nppGetMaxThreadsPerBlock (void)`
Get the maximum number of threads per block on the active CUDA device.
- `int nppGetMaxThreadsPerSM (void)`
Get the maximum number of threads per SM for the active GPU.
- `const char * nppGetGpuName (void)`
Get the name of the active CUDA device.
- `cudaStream_t nppGetStream (void)`
Get the NPP CUDA stream.
- `void nppSetStream (cudaStream_t hStream)`
Set the NPP CUDA stream.

7.1.1 Detailed Description

Basic functions for library management, in particular library version and device property query functions.

7.1.2 Function Documentation

7.1.2.1 `NppGpuComputeCapability nppGetGpuComputeCapability (void)`

What CUDA compute model is supported by the active CUDA device?

Before trying to call any NPP functions, the user should make a call this function to ensure that the current machine has a CUDA capable device.

Returns:

An enum value representing if a CUDA capable device was found and what level of compute capabilities it supports.

7.1.2.2 `const char* nppGetGpuName (void)`

Get the name of the active CUDA device.

Returns:

Name string of the active graphics-card/compute device in a system.

7.1.2.3 `int nppGetGpuNumSMs (void)`

Get the number of Streaming Multiprocessors (SM) on the active CUDA device.

Returns:

Number of SMs of the default CUDA device.

7.1.2.4 `const NppLibraryVersion* nppGetLibVersion (void)`

Get the NPP library version.

Returns:

A struct containing separate values for major and minor revision and build number.

7.1.2.5 `int nppGetMaxThreadsPerBlock (void)`

Get the maximum number of threads per block on the active CUDA device.

Returns:

Maximum number of threads per block on the active CUDA device.

7.1.2.6 int nppGetMaxThreadsPerSM (void)

Get the maximum number of threads per SM for the active GPU.

Returns:

Maximum number of threads per SM for the active GPU

7.1.2.7 cudaStream_t nppGetStream (void)

Get the NPP CUDA stream.

NPP enables concurrent device tasks via a global stream state variable. The NPP stream by default is set to stream 0, i.e. non-concurrent mode. A user can set the NPP stream to any valid CUDA stream. All CUDA commands issued by NPP (e.g. kernels launched by the NPP library) are then issued to that NPP stream.

7.1.2.8 void nppSetStream (cudaStream_t *hStream*)

Set the NPP CUDA stream.

See also:

[nppGetStream\(\)](#)

7.2 NPP Type Definitions and Constants

Data Structures

- struct [NppLibraryVersion](#)
- struct [NppiPoint](#)
2D Point
- struct [NppiSize](#)
2D Size This struct typically represents the size of a rectangular region in two space.
- struct [NppiRect](#)
2D Rectangle This struct contains position and size information of a rectangle in two space.
- struct [NppiHaarClassifier_32f](#)
- struct [NppiHaarBuffer](#)

Modules

- [Basic NPP Data Types](#)

Defines

- #define [NPP_MIN_8U](#) (0)
Minimum 8-bit unsigned integer.
- #define [NPP_MAX_8U](#) (255)
Maximum 8-bit unsigned integer.
- #define [NPP_MIN_16U](#) (0)
Minimum 16-bit unsigned integer.
- #define [NPP_MAX_16U](#) (65535)
Maximum 16-bit unsigned integer.
- #define [NPP_MIN_32U](#) (0)
Minimum 32-bit unsigned integer.
- #define [NPP_MAX_32U](#) (4294967295)
Maximum 32-bit unsigned integer.
- #define [NPP_MIN_64U](#) (0)
Minimum 64-bit unsigned integer.
- #define [NPP_MAX_64U](#) (18446744073709551615ULL)
Maximum 64-bit unsigned integer.
- #define [NPP_MIN_8S](#) (-128)
Minimum 8-bit signed integer.

- #define `NPP_MAX_8S` (127)
Maximum 8-bit signed integer.
- #define `NPP_MIN_16S` (-32768)
Minimum 16-bit signed integer.
- #define `NPP_MAX_16S` (32767)
Maximum 16-bit signed integer.
- #define `NPP_MIN_32S` (-2147483647 - 1)
Minimum 32-bit signed integer.
- #define `NPP_MAX_32S` (2147483647)
Maximum 32-bit signed integer.
- #define `NPP_MAX_64S` (9223372036854775807LL)
Minimum 64-bit signed integer.
- #define `NPP_MIN_64S` (-9223372036854775807LL - 1)
Maximum 64-bit signed integer.
- #define `NPP_MINABS_32F` (1.175494351e-38f)
Smallest positive 32-bit floating point value.
- #define `NPP_MAXABS_32F` (3.402823466e+38f)
Largest positive 32-bit floating point value.
- #define `NPP_MINABS_64F` (2.2250738585072014e-308)
Smallest positive 64-bit floating point value.
- #define `NPP_MAXABS_64F` (1.7976931348623158e+308)
Largest positive 64-bit floating point value.

Enumerations

- enum `NppiInterpolationMode` {
`NPPI_INTER_UNDEFINED` = 0,
`NPPI_INTER_NN` = 1,
`NPPI_INTER_LINEAR` = 2,
`NPPI_INTER_CUBIC` = 4,
`NPPI_INTER_SUPER` = 8,
`NPPI_INTER_LANCZOS` = 16,
`NPPI_SMOOTH_EDGE` = (1 << 31) }
Filtering methods.

```

• enum NppStatus {
    NPP_NOT_SUPPORTED_MODE_ERROR = -9999,
    NPP_ROUND_MODE_NOT_SUPPORTED_ERROR = -213,
    NPP_RESIZE_NO_OPERATION_ERROR = -50,
    NPP_NOT_SUFFICIENT_COMPUTE_CAPABILITY = -27,
    NPP_BAD_ARG_ERROR = -26,
    NPP_LUT_NUMBER_OF_LEVELS_ERROR = -25,
    NPP_TEXTURE_BIND_ERROR = -24,
    NPP_COEFF_ERROR = -23,
    NPP_RECT_ERROR = -22,
    NPP_QUAD_ERROR = -21,
    NPP_WRONG_INTERSECTION_ROI_ERROR = -20,
    NPP_NOT_EVEN_STEP_ERROR = -19,
    NPP_INTERPOLATION_ERROR = -18,
    NPP_RESIZE_FACTOR_ERROR = -17,
    NPP_HAAR_CLASSIFIER_PIXEL_MATCH_ERROR = -16,
    NPP_MEMFREE_ERR = -15,
    NPP_MEMSET_ERR = -14,
    NPP_MEMCPY_ERROR = -13,
    NPP_MEM_ALLOC_ERR = -12,
    NPP_HISTO_NUMBER_OF_LEVELS_ERROR = -11,
    NPP_MIRROR_FLIP_ERR = -10,
    NPP_INVALID_INPUT = -9,
    NPP_ALIGNMENT_ERROR = -8,
    NPP_STEP_ERROR = -7,
    NPP_SIZE_ERROR = -6,
    NPP_POINTER_ERROR = -5,
    NPP_NULL_POINTER_ERROR = -4,
    NPP_CUDA_KERNEL_EXECUTION_ERROR = -3,
    NPP_NOT_IMPLEMENTED_ERROR = -2,
    NPP_ERROR = -1,
    NPP_NO_ERROR = 0,
    NPP_SUCCESS = NPP_NO_ERROR,
    NPP_WARNING = 1,
    NPP_WRONG_INTERSECTION_QUAD_WARNING = 2,
    NPP_MISALIGNED_DST_ROI_WARNING = 3,
    NPP_AFFINE_QUAD_INCORRECT_WARNING = 4,
    NPP_DOUBLE_SIZE_WARNING = 5,
    NPP_ODD_ROI_WARNING = 6,
    NPP_WRONG_INTERSECTION_ROI_WARNING = 29 }

```

Error Status Codes.

- enum `NppGpuComputeCapability` {
`NPP_CUDA_UNKNOWN_VERSION` = -1,
`NPP_CUDA_NOT_CAPABLE`,
`NPP_CUDA_1_0`,
`NPP_CUDA_1_1`,
`NPP_CUDA_1_2`,
`NPP_CUDA_1_3`,
`NPP_CUDA_2_0` }
- enum `NppiAxis` {
`NPP_HORIZONTAL_AXIS`,
`NPP_VERTICAL_AXIS`,
`NPP_BOTH_AXIS` }
- enum `NppCmpOp` {
`NPP_CMP_LESS`,
`NPP_CMP_LESS_EQ`,
`NPP_CMP_EQ`,
`NPP_CMP_GREATER_EQ`,
`NPP_CMP_GREATER` }
- enum `NppRoundMode` {
`NPP_RND_ZERO`,
`NPP_RND_NEAR`,
`NPP_RND_FINANCIAL` }
- enum `NppiBorderType` {
`NPP_BORDER_UNDEFINED` = -1,
`NPP_BORDER_NONE` = `NPP_BORDER_UNDEFINED`,
`NPP_BORDER_CONSTANT` = 0,
`NPP_BORDER_REPLICATE` = 1,
`NPP_BORDER_WRAP` = 2 }
- enum `NppHintAlgorithm` {
`nppAlgHintNone`,
`nppAlgHintFast`,
`nppAlgHintAccurate` }

7.2.1 Define Documentation

7.2.1.1 #define NPP_MAX_16S (32767)

Maximum 16-bit signed integer.

7.2.1.2 #define NPP_MAX_16U (65535)

Maximum 16-bit unsigned integer.

7.2.1.3 #define NPP_MAX_32S (2147483647)

Maximum 32-bit signed integer.

7.2.1.4 #define NPP_MAX_32U (4294967295)

Maximum 32-bit unsigned integer.

7.2.1.5 #define NPP_MAX_64S (9223372036854775807LL)

Minimum 64-bit signed integer.

7.2.1.6 #define NPP_MAX_64U (18446744073709551615ULL)

Maximum 64-bit unsigned integer.

7.2.1.7 #define NPP_MAX_8S (127)

Maximum 8-bit signed integer.

7.2.1.8 #define NPP_MAX_8U (255)

Maximum 8-bit unsigned integer.

7.2.1.9 #define NPP_MAXABS_32F (3.402823466e+38f)

Largest positive 32-bit floating point value.

7.2.1.10 #define NPP_MAXABS_64F (1.7976931348623158e+308)

Largest positive 64-bit floating point value.

7.2.1.11 #define NPP_MIN_16S (-32768)

Minimum 16-bit signed integer.

7.2.1.12 #define NPP_MIN_16U (0)

Minimum 16-bit unsigned integer.

7.2.1.13 #define NPP_MIN_32S (-2147483647 - 1)

Minimum 32-bit signed integer.

7.2.1.14 #define NPP_MIN_32U (0)

Minimum 32-bit unsigned integer.

7.2.1.15 #define NPP_MIN_64S (-9223372036854775807LL - 1)

Maximum 64-bit signed integer.

7.2.1.16 #define NPP_MIN_64U (0)

Minimum 64-bit unsigned integer.

7.2.1.17 #define NPP_MIN_8S (-128)

Minimum 8-bit signed integer.

7.2.1.18 #define NPP_MIN_8U (0)

Minimum 8-bit unsigned integer.

7.2.1.19 #define NPP_MINABS_32F (1.175494351e-38f)

Smallest positive 32-bit floating point value.

7.2.1.20 #define NPP_MINABS_64F (2.2250738585072014e-308)

Smallest positive 64-bit floating point value.

7.2.2 Enumeration Type Documentation**7.2.2.1 enum NppCmpOp**

Enumerator:

NPP_CMP_LESS

NPP_CMP_LESS_EQ

NPP_CMP_EQ

NPP_CMP_GREATER_EQ

NPP_CMP_GREATER

7.2.2.2 enum NppGpuComputeCapability

Enumerator:

NPP_CUDA_UNKNOWN_VERSION Indicates that the compute-capability query failed.

NPP_CUDA_NOT_CAPABLE Indicates that no CUDA capable device was found on machine.

NPP_CUDA_1_0 Indicates that CUDA 1.0 capable device is default device on machine.

NPP_CUDA_1_1 Indicates that CUDA 1.1 capable device.

NPP_CUDA_1_2 Indicates that CUDA 1.2 capable device.

NPP_CUDA_1_3 Indicates that CUDA 1.3 capable device.

NPP_CUDA_2_0 Indicates that CUDA 2.0 or better is default device on machine.

7.2.2.3 enum NppHintAlgorithm

Enumerator:

nppAlgHintNone

nppAlgHintFast

nppAlgHintAccurate

7.2.2.4 enum NppiAxis

Enumerator:

NPP_HORIZONTAL_AXIS

NPP_VERTICAL_AXIS

NPP_BOTH_AXIS

7.2.2.5 enum NppiBorderType

Enumerator:

NPP_BORDER_UNDEFINED

NPP_BORDER_NONE

NPP_BORDER_CONSTANT

NPP_BORDER_REPLICATE

NPP_BORDER_WRAP

7.2.2.6 enum NppiInterpolationMode

Filtering methods.

Enumerator:

NPPI_INTER_UNDEFINED

NPPI_INTER_NN Nearest neighbor filtering.

NPPI_INTER_LINEAR Linear interpolation.

NPPI_INTER_CUBIC Cubic interpolation.

NPPI_INTER_SUPER Super sampling.

NPPI_INTER_LANCZOS Lanczos filtering.

NPPI_SMOOTH_EDGE Smooth edge filtering.

7.2.2.7 enum NppRoundMode

Enumerator:

NPP_RND_ZERO
NPP_RND_NEAR
NPP_RND_FINANCIAL

7.2.2.8 enum NppStatus

Error Status Codes.

Almost all NPP function return error-status information using these return codes. Negative return codes indicate errors, positive return codes indicate warnings, a return code of 0 indicates success.

Enumerator:

NPP_NOT_SUPPORTED_MODE_ERROR
NPP_ROUND_MODE_NOT_SUPPORTED_ERROR
NPP_RESIZE_NO_OPERATION_ERROR
NPP_NOT_SUFFICIENT_COMPUTE_CAPABILITY
NPP_BAD_ARG_ERROR
NPP_LUT_NUMBER_OF_LEVELS_ERROR
NPP_TEXTURE_BIND_ERROR
NPP_COEFF_ERROR
NPP_RECT_ERROR
NPP_QUAD_ERROR
NPP_WRONG_INTERSECTION_ROI_ERROR
NPP_NOT_EVEN_STEP_ERROR
NPP_INTERPOLATION_ERROR
NPP_RESIZE_FACTOR_ERROR
NPP_HAAR_CLASSIFIER_PIXEL_MATCH_ERROR
NPP_MEMFREE_ERR
NPP_MEMSET_ERR
NPP_MEMCPY_ERROR
NPP_MEM_ALLOC_ERR
NPP_HISTO_NUMBER_OF_LEVELS_ERROR
NPP_MIRROR_FLIP_ERR
NPP_INVALID_INPUT
NPP_ALIGNMENT_ERROR
NPP_STEP_ERROR Step is less or equal zero.
NPP_SIZE_ERROR
NPP_POINTER_ERROR
NPP_NULL_POINTER_ERROR
NPP_CUDA_KERNEL_EXECUTION_ERROR

NPP_NOT_IMPLEMENTED_ERROR

NPP_ERROR

NPP_NO_ERROR Error free operation.

NPP_SUCCESS Successful operation (same as *NPP_NO_ERROR*).

NPP_WARNING

NPP_WRONG_INTERSECTION_QUAD_WARNING

NPP_MISALIGNED_DST_ROI_WARNING Speed reduction due to uncoalesced memory accesses warning.

NPP_AFFINE_QUAD_INCORRECT_WARNING Indicates that the quadrangle passed to one of affine warping functions doesn't have necessary properties. First 3 vertices are used, the fourth vertex discarded.

NPP_DOUBLE_SIZE_WARNING Indicates that in case of 422/411/420 sampling the ROI width/height was modified for proper processing.

NPP_ODD_ROI_WARNING Indicates that for 422/411/420 sampling the ROI width/height was forced to even value.

NPP_WRONG_INTERSECTION_ROI_WARNING ROI doesn't intersect source or destination ROI/image. No operation performed.

7.3 Basic NPP Data Types

Data Structures

- struct [Npp16sc](#)
Complex Number This struct represents a short complex number.
- struct [Npp32sc](#)
Complex Number This struct represents a signed int complex number.
- struct [Npp32fc](#)
Complex Number This struct represents a single floating-point complex number.
- struct [Npp64sc](#)
Complex Number This struct represents a long long complex number.
- struct [Npp64fc](#)
Complex Number This struct represents a double floating-point complex number.

Typedefs

- typedef unsigned char [Npp8u](#)
8-bit unsigned chars
- typedef signed char [Npp8s](#)
8-bit signed chars
- typedef unsigned short [Npp16u](#)
16-bit unsigned integers
- typedef short [Npp16s](#)
16-bit signed integers
- typedef unsigned int [Npp32u](#)
32-bit unsigned integers
- typedef int [Npp32s](#)
32-bit signed integers
- typedef unsigned long long [Npp64u](#)
64-bit unsigned integers
- typedef long long [Npp64s](#)
64-bit signed integers
- typedef float [Npp32f](#)
32-bit (IEEE) floating-point numbers

- typedef double [Npp64f](#)
64-bit floating-point numbers

7.3.1 Typedef Documentation

7.3.1.1 typedef short Npp16s

16-bit signed integers

7.3.1.2 typedef unsigned short Npp16u

16-bit unsigned integers

7.3.1.3 typedef float Npp32f

32-bit (IEEE) floating-point numbers

7.3.1.4 typedef int Npp32s

32-bit signed integers

7.3.1.5 typedef unsigned int Npp32u

32-bit unsigned integers

7.3.1.6 typedef double Npp64f

64-bit floating-point numbers

7.3.1.7 typedef long long Npp64s

64-bit signed integers

7.3.1.8 typedef unsigned long long Npp64u

64-bit unsigned integers

7.3.1.9 typedef signed char Npp8s

8-bit signed chars

7.3.1.10 typedef unsigned char Npp8u

8-bit unsigned chars

7.4 NPP Image Processing

Modules

- [Memory Management](#)
Routines for allocating and deallocating pitched image storage.
- [Data-Exchange and Initialization](#)
Primitives for initialization, copying and converting image data.
- [Arithmetic and Logical Operations](#)
- [Threshold and Compare Operations](#)
Methods for pixel-wise threshold and compare operations.
- [Statistics Functions](#)
Routines computing statistical image information.
- [Filtering Functions](#)
Linear and non-linear image filtering functions.
- [Morphological Operations](#)
Morphological image operations.
- [Image Linear Transforms](#)
Linear image transformations.
- [Compression](#)
Image compression primitives.
- [Geometric Transforms](#)
Routines manipulating an image's geometry.
- [Color Conversion](#)
Image color space and sample conversion operations.
- [Labeling and Segmentation](#)
Pixel labeling and image segmentation operations.

Enumerations

- [enum NppiAlphaOp](#) {
 [NPPI_OP_ALPHA_OVER](#),
 [NPPI_OP_ALPHA_IN](#),
 [NPPI_OP_ALPHA_OUT](#),
 [NPPI_OP_ALPHA_ATOP](#),
 [NPPI_OP_ALPHA_XOR](#),
 [NPPI_OP_ALPHA_PLUS](#),

```
NPPI_OP_ALPHA_OVER_PREMUL,  
NPPI_OP_ALPHA_IN_PREMUL,  
NPPI_OP_ALPHA_OUT_PREMUL,  
NPPI_OP_ALPHA_ATOP_PREMUL,  
NPPI_OP_ALPHA_XOR_PREMUL,  
NPPI_OP_ALPHA_PLUS_PREMUL,  
NPPI_OP_ALPHA_PREMUL }
```

7.4.1 Enumeration Type Documentation

7.4.1.1 enum NppiAlphaOp

Enumerator:

```
NPPI_OP_ALPHA_OVER  
NPPI_OP_ALPHA_IN  
NPPI_OP_ALPHA_OUT  
NPPI_OP_ALPHA_ATOP  
NPPI_OP_ALPHA_XOR  
NPPI_OP_ALPHA_PLUS  
NPPI_OP_ALPHA_OVER_PREMUL  
NPPI_OP_ALPHA_IN_PREMUL  
NPPI_OP_ALPHA_OUT_PREMUL  
NPPI_OP_ALPHA_ATOP_PREMUL  
NPPI_OP_ALPHA_XOR_PREMUL  
NPPI_OP_ALPHA_PLUS_PREMUL  
NPPI_OP_ALPHA_PREMUL
```


7.5 Memory Management

Routines for allocating and deallocating pitched image storage.

Functions

- void [nppiFree](#) (void *pData)
Free method for any 2D allocated memory.

Image-Memory Allocation

ImageAllocator methods for 2D arrays of data.

The allocators have width and height parameters to specify the size of the image data being allocated. They return a pointer to the newly created memory and return the numbers of bytes between successive lines.

If the memory allocation failed due to lack of free device memory or device memory fragmentation the routine returns 0.

All allocators return memory with line strides that are beneficial for performance. It is not mandatory to use these allocators. Any valid CUDA device-memory pointers can be used by the NPP primitives and there are no restrictions on line strides.

- [Npp8u * nppiMalloc_8u_C1](#) (int nWidthPixels, int nHeightPixels, int *pStepBytes)
8-bit unsigned image memory allocator.
- [Npp8u * nppiMalloc_8u_C2](#) (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 8-bit unsigned image memory allocator.
- [Npp8u * nppiMalloc_8u_C3](#) (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 8-bit unsigned image memory allocator.
- [Npp8u * nppiMalloc_8u_C4](#) (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 8-bit unsigned image memory allocator.
- [Npp16u * nppiMalloc_16u_C1](#) (int nWidthPixels, int nHeightPixels, int *pStepBytes)
16-bit unsigned image memory allocator.
- [Npp16u * nppiMalloc_16u_C2](#) (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 16-bit unsigned image memory allocator.
- [Npp16u * nppiMalloc_16u_C3](#) (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 16-bit unsigned image memory allocator.
- [Npp16u * nppiMalloc_16u_C4](#) (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 16-bit unsigned image memory allocator.
- [Npp16s * nppiMalloc_16s_C1](#) (int nWidthPixels, int nHeightPixels, int *pStepBytes)
16-bit signed image memory allocator.

- `Npp16s * nppiMalloc_16s_C2` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 16-bit signed image memory allocator.
- `Npp16s * nppiMalloc_16s_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 16-bit signed image memory allocator.
- `Npp16sc * nppiMalloc_16sc_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
1 channel 16-bit signed complex image memory allocator.
- `Npp16sc * nppiMalloc_16sc_C2` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 16-bit signed complex image memory allocator.
- `Npp16sc * nppiMalloc_16sc_C3` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 16-bit signed complex image memory allocator.
- `Npp16sc * nppiMalloc_16sc_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 16-bit signed complex image memory allocator.
- `Npp32s * nppiMalloc_32s_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
32-bit signed image memory allocator.
- `Npp32s * nppiMalloc_32s_C3` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 32-bit signed image memory allocator.
- `Npp32s * nppiMalloc_32s_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 32-bit signed image memory allocator.
- `Npp32sc * nppiMalloc_32sc_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
32-bit integer complex image memory allocator.
- `Npp32sc * nppiMalloc_32sc_C2` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 32-bit integer complex image memory allocator.
- `Npp32sc * nppiMalloc_32sc_C3` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 32-bit integer complex image memory allocator.
- `Npp32sc * nppiMalloc_32sc_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 32-bit integer complex image memory allocator.
- `Npp32f * nppiMalloc_32f_C1` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
32-bit floating point image memory allocator.
- `Npp32f * nppiMalloc_32f_C2` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
2 channel 32-bit floating point image memory allocator.
- `Npp32f * nppiMalloc_32f_C3` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
3 channel 32-bit floating point image memory allocator.
- `Npp32f * nppiMalloc_32f_C4` (int nWidthPixels, int nHeightPixels, int *pStepBytes)
4 channel 32-bit floating point image memory allocator.

- `Npp32fc * nppiMalloc_32fc_C1` (int *nWidthPixels*, int *nHeightPixels*, int **pStepBytes*)
32-bit float complex image memory allocator.
- `Npp32fc * nppiMalloc_32fc_C2` (int *nWidthPixels*, int *nHeightPixels*, int **pStepBytes*)
2 channel 32-bit float complex image memory allocator.
- `Npp32fc * nppiMalloc_32fc_C3` (int *nWidthPixels*, int *nHeightPixels*, int **pStepBytes*)
3 channel 32-bit float complex image memory allocator.
- `Npp32fc * nppiMalloc_32fc_C4` (int *nWidthPixels*, int *nHeightPixels*, int **pStepBytes*)
4 channel 32-bit float complex image memory allocator.

7.5.1 Detailed Description

Routines for allocating and deallocating pitched image storage.

These methods are provided for convenience. They allocate memory that may contain additional padding bytes at the end of each line of pixels. Though padding is not necessary for any of the NPP image-processing primitives to work correctly, its absence may cause severe performance degradation compared to properly padded images.

7.5.2 Function Documentation

7.5.2.1 void nppiFree (void * *pData*)

Free method for any 2D allocated memory.

This method should be used to free memory allocated with any of the `nppiMalloc_<modifier>` methods.

Parameters:

pData A pointer to memory allocated using `nppiMalloc_<modifier>`.

7.5.2.2 Npp16s* nppiMalloc_16s_C1 (int *nWidthPixels*, int *nHeightPixels*, int **pStepBytes*)

16-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes Line Step.

Returns:

Pointer to new image data.

7.5.2.3 Npp16s* nppiMalloc_16s_C2 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

2 channel 16-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.4 Npp16s* nppiMalloc_16s_C4 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

4 channel 16-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.5 Npp16sc* nppiMalloc_16sc_C1 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

1 channel 16-bit signed complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.6 Npp16sc* nppiMalloc_16sc_C2 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

2 channel 16-bit signed complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.7 Npp16sc* nppiMalloc_16sc_C3 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

3 channel 16-bit signed complex image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.8 Npp16sc* nppiMalloc_16sc_C4 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

4 channel 16-bit signed complex image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.9 Npp16u* nppiMalloc_16u_C1 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

16-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.10 Npp16u* nppiMalloc_16u_C2 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

2 channel 16-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.11 Npp16u* nppiMalloc_16u_C3 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

3 channel 16-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.12 Npp16u* nppiMalloc_16u_C4 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

4 channel 16-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.13 Npp32f* nppiMalloc_32f_C1 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

32-bit floating point image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.14 Npp32f* nppiMalloc_32f_C2 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

2 channel 32-bit floating point image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.15 Npp32f* nppiMalloc_32f_C3 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

3 channel 32-bit floating point image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.16 Npp32f* nppiMalloc_32f_C4 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

4 channel 32-bit floating point image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.17 Npp32fc* nppiMalloc_32fc_C1 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

32-bit float complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.18 Npp32fc* nppiMalloc_32fc_C2 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

2 channel 32-bit float complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.19 Npp32fc* nppiMalloc_32fc_C3 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

3 channel 32-bit float complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.
pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.20 Npp32fc* nppiMalloc_32fc_C4 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

4 channel 32-bit float complex image memory allocator.

Parameters:

nWidthPixels Image width.
nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.21 Npp32s* nppiMalloc_32s_C1 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

32-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.22 Npp32s* nppiMalloc_32s_C3 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

3 channel 32-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.23 Npp32s* nppiMalloc_32s_C4 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

4 channel 32-bit signed image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.24 Npp32sc* nppiMalloc_32sc_C1 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

32-bit integer complex image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.25 Npp32sc* nppiMalloc_32sc_C2 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

2 channel 32-bit integer complex image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.26 Npp32sc* nppiMalloc_32sc_C3 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

3 channel 32-bit integer complex image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.27 Npp32sc* nppiMalloc_32sc_C4 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

4 channel 32-bit integer complex image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.28 Npp8u* nppiMalloc_8u_C1 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

8-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.29 Npp8u* nppiMalloc_8u_C2 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

2 channel 8-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.30 Npp8u* nppiMalloc_8u_C3 (int nWidthPixels, int nHeightPixels, int * pStepBytes)

3 channel 8-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.5.2.31 Npp8u* nppiMalloc_8u_C4 (int *nWidthPixels*, int *nHeightPixels*, int * *pStepBytes*)

4 channel 8-bit unsigned image memory allocator.

Parameters:

nWidthPixels Image width.

nHeightPixels Image height.

pStepBytes [Line Step](#).

Returns:

Pointer to new image data.

7.6 Data-Exchange and Initialization

Primitives for initialization, copying and converting image data.

Image-Memory Set

Set methods for images of various types.

Images are passed to these primitives via a pointer to the image data (first pixel in the ROI) and a step-width, i.e. the number of bytes between successive lines. The size of the area to be set (region-of-interest, ROI) is specified via a [NppiSize](#) struct. In addition to the image data and ROI, all methods have a parameter to specify the value being set. In case of single channel images this is a single value, in case of multi-channel, an array of values is passed.

- [NppStatus nppiSet_8s_C1R](#) ([Npp8s](#) nValue, [Npp8s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
8-bit image set.
- [NppStatus nppiSet_8s_C2R](#) ([Npp8s](#) aValue[2], [Npp8s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
8-bit two-channel image set.
- [NppStatus nppiSet_8s_C3R](#) ([Npp8s](#) aValue[3], [Npp8s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
8-bit three-channel image set.
- [NppStatus nppiSet_8s_C4R](#) ([Npp8s](#) aValue[4], [Npp8s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
8-bit four-channel image set.
- [NppStatus nppiSet_8s_AC4R](#) ([Npp8s](#) aValue[3], [Npp8s](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
8-bit four-channel image set ignoring alpha channel.
- [NppStatus nppiSet_8u_C1R](#) ([Npp8u](#) nValue, [Npp8u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
8-bit unsigned image set.
- [NppStatus nppiSet_8u_C1MR](#) ([Npp8u](#) nValue, [Npp8u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const [Npp8u](#) *pMask, int nMaskStep)
Masked 8-bit unsigned image set.
- [NppStatus nppiSet_8u_C4R](#) (const [Npp8u](#) aValues[4], [Npp8u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
4 channel 8-bit unsigned image set.
- [NppStatus nppiSet_8u_C4MR](#) (const [Npp8u](#) aValues[4], [Npp8u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const [Npp8u](#) *pMask, int nMaskStep)
Masked 4 channel 8-bit unsigned image set.
- [NppStatus nppiSet_8u_AC4R](#) (const [Npp8u](#) aValues[3], [Npp8u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI)
4 channel 8-bit unsigned image set method, not affecting Alpha channel.
- [NppStatus nppiSet_8u_AC4MR](#) (const [Npp8u](#) aValues[3], [Npp8u](#) *pDst, int nDstStep, [NppiSize](#) oSizeROI, const [Npp8u](#) *pMask, int nMaskStep)

Masked 4 channel 8-bit unsigned image set method, not affecting Alpha channel.

- **NppStatus nppiSet_8u_C4CR** (**Npp8u** nValue, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 8-bit unsigned image set affecting only single channel.
- **NppStatus nppiSet_16u_C1R** (**Npp16u** nValue, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
16-bit unsigned image set.
- **NppStatus nppiSet_16u_C1MR** (**Npp16u** nValue, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, int nMaskStep)
Masked 16-bit unsigned image set.
- **NppStatus nppiSet_16u_C2R** (const **Npp16u** aValues[2], **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
2 channel 16-bit unsigned image set.
- **NppStatus nppiSet_16u_C4R** (const **Npp16u** aValues[4], **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 16-bit unsigned image set.
- **NppStatus nppiSet_16u_C4MR** (const **Npp16u** aValues[4], **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, int nMaskStep)
Masked 4 channel 16-bit unsigned image set.
- **NppStatus nppiSet_16u_AC4R** (const **Npp16u** aValues[3], **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 16-bit unsigned image set method, not affecting Alpha channel.
- **NppStatus nppiSet_16u_AC4MR** (const **Npp16u** aValues[3], **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, int nMaskStep)
Masked 4 channel 16-bit unsigned image set method, not affecting Alpha channel.
- **NppStatus nppiSet_16u_C4CR** (**Npp16u** nValue, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 16-bit unsigned image set affecting only single channel.
- **NppStatus nppiSet_16s_C1R** (**Npp16s** nValue, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)
16-bit image set.
- **NppStatus nppiSet_16s_C1MR** (**Npp16s** nValue, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, int nMaskStep)
Masked 16-bit image set.
- **NppStatus nppiSet_16s_C2R** (const **Npp16s** aValues[2], **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)
2 channel 16-bit image set.
- **NppStatus nppiSet_16s_C4R** (const **Npp16s** aValues[4], **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 16-bit image set.

- `NppStatus nppiSet_16s_C4MR` (const `Npp16s` aValues[4], `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 4 channel 16-bit image set.
- `NppStatus nppiSet_16s_AC4R` (const `Npp16s` aValues[3], `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 16-bit image set method, not affecting Alpha channel.
- `NppStatus nppiSet_16s_AC4MR` (const `Npp16s` aValues[3], `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 4 channel 16-bit image set method, not affecting Alpha channel.
- `NppStatus nppiSet_16s_C4CR` (`Npp16s` nValue, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 16-bit unsigned image set affecting only single channel.
- `NppStatus nppiSet_16sc_C1R` (`Npp16sc` oValue, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
16-bit complex integer image set.
- `NppStatus nppiSet_16sc_C2R` (`Npp16sc` aValue[2], `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
16-bit complex integer two-channel image set.
- `NppStatus nppiSet_16sc_C3R` (`Npp16sc` aValue[3], `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
16-bit complex integer three-channel image set.
- `NppStatus nppiSet_16sc_AC4R` (`Npp16sc` aValue[3], `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
16-bit complex integer four-channel image set ignoring alpha.
- `NppStatus nppiSet_16sc_C4R` (`Npp16sc` aValue[4], `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
16-bit complex integer four-channel image set.
- `NppStatus nppiSet_32s_C1R` (`Npp32s` nValue, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
32-bit image set.
- `NppStatus nppiSet_32s_C1MR` (`Npp32s` nValue, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 32-bit image set.
- `NppStatus nppiSet_32s_C4R` (const `Npp32s` aValues[4], `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 32-bit image set.
- `NppStatus nppiSet_32s_C4MR` (const `Npp32s` aValues[4], `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked 4 channel 32-bit image set.

- **NppStatus nppiSet_32s_AC4R** (const **Npp32s** aValues[3], **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 16-bit image set method, not affecting Alpha channel.
- **NppStatus nppiSet_32s_AC4MR** (const **Npp32s** aValues[3], **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, int nMaskStep)
Masked 4 channel 16-bit image set method, not affecting Alpha channel.
- **NppStatus nppiSet_32s_C4CR** (**Npp32s** nValue, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 32-bit unsigned image set affecting only single channel.
- **NppStatus nppiSet_32sc_C1R** (**Npp32sc** oValue, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Single channel 32-bit complex integer image set.
- **NppStatus nppiSet_32sc_C2R** (**Npp32sc** aValue[2], **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Two channel 32-bit complex integer image set.
- **NppStatus nppiSet_32sc_C3R** (**Npp32sc** aValue[3], **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three channel 32-bit complex integer image set.
- **NppStatus nppiSet_32sc_C4R** (**Npp32sc** aValue[4], **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four channel 32-bit complex integer image set.
- **NppStatus nppiSet_32sc_AC4R** (**Npp32sc** aValue[3], **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI)
32-bit complex integer four-channel image set ignoring alpha.
- **NppStatus nppiSet_32f_C1R** (**Npp32f** nValue, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
32-bit floating point image set.
- **NppStatus nppiSet_32f_C1MR** (**Npp32f** nValue, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, int nMaskStep)
Masked 32-bit floating point image set.
- **NppStatus nppiSet_32f_C4R** (const **Npp32f** aValues[4], **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 32-bit floating point image set.
- **NppStatus nppiSet_32f_C4MR** (const **Npp32f** aValues[4], **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, int nMaskStep)
Masked 4 channel 32-bit floating point image set.
- **NppStatus nppiSet_32f_AC4R** (const **Npp32f** aValues[3], **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 32-bit floating point image set method, not affecting Alpha channel.

- **NppStatus nppiSet_32f_AC4MR** (const **Npp32f** aValues[3], **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, int nMaskStep)
Masked 4 channel 32-bit floating point image set method, not affecting Alpha channel.
- **NppStatus nppiSet_32f_C4CR** (**Npp32f** nValue, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 32-bit floating point image set affecting only single channel.
- **NppStatus nppiSet_32fc_C1R** (**Npp32fc** oValue, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Single channel 32-bit complex image set.
- **NppStatus nppiSet_32fc_C2R** (**Npp32fc** aValue[2], **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Two channel 32-bit complex image set.
- **NppStatus nppiSet_32fc_C3R** (**Npp32fc** aValue[3], **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three channel 32-bit complex image set.
- **NppStatus nppiSet_32fc_C4R** (**Npp32fc** aValue[4], **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four channel 32-bit complex image set.
- **NppStatus nppiSet_32fc_AC4R** (**Npp32fc** aValue[3], **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
32-bit complex four-channel image set ignoring alpha.

Image-Memory Copy

Copy methods for images of various types.

In addition to routines for copying pixels of identical layout from one image to another, there are copy routines for select channels as well as packed-planar conversions:

- Select channel to multi-channel copy. E.g. given a three-channel source and destination image one may copy the second channel of the source to the third channel of the destination.
- Single channel to multi-channel copy. E.g. given a single-channel source and a four-channel destination, one may copy the contents of the single-channel source to the second channel of the destination.
- Select channel to single-channel copy. E.g. given a three-channel source and a single-channel destination one may copy the third channel of the source to the destination.
- Multi-channel to planar copy. These copy operations split a multi-channel image into a set of single-channel images.
- Planar image to multi-channel copy. These copy routines combine separate color-planes (single channel images) into a single multi-channel image.
- **NppStatus nppiCopy_8s_C1R** (const **Npp8s** *pSrc, int nSrcStep, **Npp8s** *pDst, int nDstStep, **NppiSize** oSizeROI)
8-bit image copy.

- `NppStatus nppiCopy_8s_C2R` (const `Npp8s` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Two-channel 8-bit image copy.
- `NppStatus nppiCopy_8s_C3R` (const `Npp8s` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three-channel 8-bit image copy.
- `NppStatus nppiCopy_8s_C4R` (const `Npp8s` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel 8-bit image copy.
- `NppStatus nppiCopy_8s_AC4R` (const `Npp8s` *pSrc, int nSrcStep, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel 8-bit image copy, ignoring alpha channel.
- `NppStatus nppiCopy_8u_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
8-bit unsigned image copy.
- `NppStatus nppiCopy_8u_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 8-bit unsigned image copy.
- `NppStatus nppiCopy_8u_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 8-bit unsigned image copy, not affecting Alpha channel.
- `NppStatus nppiCopy_16u_C1R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
16-bit unsigned image copy.
- `NppStatus nppiCopy_16u_C4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 16-bit unsigned image copy.
- `NppStatus nppiCopy_16u_AC4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 16-bit unsigned image copy, not affecting Alpha channel.
- `NppStatus nppiCopy_16s_C1R` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
16-bit image copy.
- `NppStatus nppiCopy_16s_C4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 16-bit image copy.
- `NppStatus nppiCopy_16s_AC4R` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)

4 channel 16-bit image copy, not affecting Alpha.

- `NppStatus nppiCopy_16sc_C1R` (const `Npp16sc` *pSrc, int nSrcStep, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

16-bit complex image copy.

- `NppStatus nppiCopy_16sc_C2R` (const `Npp16sc` *pSrc, int nSrcStep, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Two-channel 16-bit complex image copy.

- `NppStatus nppiCopy_16sc_C3R` (const `Npp16sc` *pSrc, int nSrcStep, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three-channel 16-bit complex image copy.

- `NppStatus nppiCopy_16sc_C4R` (const `Npp16sc` *pSrc, int nSrcStep, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four-channel 16-bit complex image copy.

- `NppStatus nppiCopy_16sc_AC4R` (const `Npp16sc` *pSrc, int nSrcStep, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four-channel 16-bit complex image copy, ignoring alpha.

- `NppStatus nppiCopy_32s_C1R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

32-bit image copy.

- `NppStatus nppiCopy_32s_C4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

4 channel 32-bit image copy.

- `NppStatus nppiCopy_32s_AC4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

4 channel 32-bit image copy, not affecting Alpha.

- `NppStatus nppiCopy_32sc_C1R` (const `Npp32sc` *pSrc, int nSrcStep, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

32-bit complex image copy.

- `NppStatus nppiCopy_32sc_C2R` (const `Npp32sc` *pSrc, int nSrcStep, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Two-channel 32-bit complex image copy.

- `NppStatus nppiCopy_32sc_C3R` (const `Npp32sc` *pSrc, int nSrcStep, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three-channel 32-bit complex image copy.

- `NppStatus nppiCopy_32sc_C4R` (const `Npp32sc` *pSrc, int nSrcStep, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four-channel 32-bit complex image copy.

- `NppStatus nppiCopy_32sc_AC4R` (const `Npp32sc` *pSrc, int nSrcStep, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel 32-bit complex image copy, ignoring alpha.
- `NppStatus nppiCopy_32f_C1R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
32-bit floating point image copy.
- `NppStatus nppiCopy_32f_C4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 32-bit floating point image copy.
- `NppStatus nppiCopy_32f_AC4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 32-bit floating point image copy, not affecting Alpha.
- `NppStatus nppiCopy_32fc_C1R` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)
32-bit floating-point complex image copy.
- `NppStatus nppiCopy_32fc_C2R` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Two-channel 32-bit floating-point complex image copy.
- `NppStatus nppiCopy_32fc_C3R` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three-channel 32-bit floating-point complex image copy.
- `NppStatus nppiCopy_32fc_C4R` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel 32-bit floating-point complex image copy.
- `NppStatus nppiCopy_32fc_AC4R` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel 32-bit floating-point complex image copy, ignoring alpha.
- `NppStatus nppiCopy_8u_C1MR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation 8-bit unsigned image copy.
- `NppStatus nppiCopy_8u_C3MR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation three channel 8-bit unsigned image copy.
- `NppStatus nppiCopy_8u_C4MR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation four channel 8-bit unsigned image copy.
- `NppStatus nppiCopy_8u_AC4MR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation four channel 8-bit unsigned image copy, ignoring alpha.

- `NppStatus nppiCopy_16u_C1MR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation 16-bit unsigned image copy.
- `NppStatus nppiCopy_16u_C3MR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation three channel 16-bit unsigned image copy.
- `NppStatus nppiCopy_16u_C4MR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation four channel 16-bit unsigned image copy.
- `NppStatus nppiCopy_16u_AC4MR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation four channel 16-bit unsigned image copy, ignoring alpha.
- `NppStatus nppiCopy_16s_C1MR` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation 16-bit signed image copy.
- `NppStatus nppiCopy_16s_C3MR` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation three channel 16-bit signed image copy.
- `NppStatus nppiCopy_16s_C4MR` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation four channel 16-bit signed image copy.
- `NppStatus nppiCopy_16s_AC4MR` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation four channel 16-bit signed image copy, ignoring alpha.
- `NppStatus nppiCopy_32s_C1MR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation 32-bit signed image copy.
- `NppStatus nppiCopy_32s_C3MR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation three channel 32-bit signed image copy.
- `NppStatus nppiCopy_32s_C4MR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation four channel 32-bit signed image copy.
- `NppStatus nppiCopy_32s_AC4MR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)
Masked Operation four channel 32-bit signed image copy, ignoring alpha.
- `NppStatus nppiCopy_32f_C1MR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation 32-bit float image copy.

- `NppStatus nppiCopy_32f_C3MR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation three channel 32-bit float image copy.

- `NppStatus nppiCopy_32f_C4MR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 32-bit float image copy.

- `NppStatus nppiCopy_32f_AC4MR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp8u` *pMask, int nMaskStep)

Masked Operation four channel 32-bit float image copy, ignoring alpha.

- `NppStatus nppiCopy_8u_C3CR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Select-channel 8-bit unsigned image copy for three-channel images.

- `NppStatus nppiCopy_8u_C4CR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Select-channel 8-bit unsigned image copy for four-channel images.

- `NppStatus nppiCopy_16s_C3CR` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Select-channel 16-bit signed image copy for three-channel images.

- `NppStatus nppiCopy_16s_C4CR` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Select-channel 16-bit signed image copy for four-channel images.

- `NppStatus nppiCopy_16u_C3CR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Select-channel 16-bit unsigned image copy for three-channel images.

- `NppStatus nppiCopy_16u_C4CR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Select-channel 16-bit unsigned image copy for four-channel images.

- `NppStatus nppiCopy_32s_C3CR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Select-channel 32-bit signed image copy for three-channel images.

- `NppStatus nppiCopy_32s_C4CR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Select-channel 32-bit signed image copy for four-channel images.

- `NppStatus nppiCopy_32f_C3CR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Select-channel 32-bit float image copy for three-channel images.

- `NppStatus nppiCopy_32f_C4CR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Select-channel 32-bit float image copy for four-channel images.
- `NppStatus nppiCopy_8u_C3C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three-channel to single-channel 8-bit unsigned image copy.
- `NppStatus nppiCopy_8u_C4C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel to single-channel 8-bit unsigned image copy.
- `NppStatus nppiCopy_16s_C3C1R` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three-channel to single-channel 16-bit signed image copy.
- `NppStatus nppiCopy_16s_C4C1R` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel to single-channel 16-bit signed image copy.
- `NppStatus nppiCopy_16u_C3C1R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three-channel to single-channel 16-bit unsigned image copy.
- `NppStatus nppiCopy_16u_C4C1R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel to single-channel 16-bit unsigned image copy.
- `NppStatus nppiCopy_32s_C3C1R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three-channel to single-channel 32-bit signed image copy.
- `NppStatus nppiCopy_32s_C4C1R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel to single-channel 32-bit signed image copy.
- `NppStatus nppiCopy_32f_C3C1R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three-channel to single-channel 32-bit float image copy.
- `NppStatus nppiCopy_32f_C4C1R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel to single-channel 32-bit float image copy.
- `NppStatus nppiCopy_8u_C1C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single-channel to three-channel 8-bit unsigned image copy.
- `NppStatus nppiCopy_8u_C1C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Single-channel to four-channel 8-bit unsigned image copy.

- **NppStatus nppiCopy_16s_C1C3R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Single-channel to three-channel 16-bit signed image copy.
- **NppStatus nppiCopy_16s_C1C4R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Single-channel to four-channel 16-bit signed image copy.
- **NppStatus nppiCopy_16u_C1C3R** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Single-channel to three-channel 16-bit unsigned image copy.
- **NppStatus nppiCopy_16u_C1C4R** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Single-channel to four-channel 16-bit unsigned image copy.
- **NppStatus nppiCopy_32s_C1C3R** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Single-channel to three-channel 32-bit signed image copy.
- **NppStatus nppiCopy_32s_C1C4R** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Single-channel to four-channel 32-bit signed image copy.
- **NppStatus nppiCopy_32f_C1C3R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Single-channel to three-channel 32-bit float image copy.
- **NppStatus nppiCopy_32f_C1C4R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Single-channel to four-channel 32-bit float image copy.
- **NppStatus nppiCopy_8u_C3P3R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *const aDst[3], int nDstStep, **NppiSize** oSizeROI)
Three-channel 8-bit unsigned packed to planar image copy.
- **NppStatus nppiCopy_8u_C4P4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *const aDst[4], int nDstStep, **NppiSize** oSizeROI)
Four-channel 8-bit unsigned packed to planar image copy.
- **NppStatus nppiCopy_16s_C3P3R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *const aDst[3], int nDstStep, **NppiSize** oSizeROI)
Three-channel 16-bit signed packed to planar image copy.
- **NppStatus nppiCopy_16s_C4P4R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *const aDst[4], int nDstStep, **NppiSize** oSizeROI)
Four-channel 16-bit signed packed to planar image copy.
- **NppStatus nppiCopy_16u_C3P3R** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *const aDst[3], int nDstStep, **NppiSize** oSizeROI)

Three-channel 16-bit unsigned packed to planar image copy.

- `NppStatus nppiCopy_16u_C4P4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *const aDst[4], int nDstStep, `NppiSize` oSizeROI)

Four-channel 16-bit unsigned packed to planar image copy.

- `NppStatus nppiCopy_32s_C3P3R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *const aDst[3], int nDstStep, `NppiSize` oSizeROI)

Three-channel 32-bit signed packed to planar image copy.

- `NppStatus nppiCopy_32s_C4P4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *const aDst[4], int nDstStep, `NppiSize` oSizeROI)

Four-channel 32-bit signed packed to planar image copy.

- `NppStatus nppiCopy_32f_C3P3R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *const aDst[3], int nDstStep, `NppiSize` oSizeROI)

Three-channel 32-bit float packed to planar image copy.

- `NppStatus nppiCopy_32f_C4P4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *const aDst[4], int nDstStep, `NppiSize` oSizeROI)

Four-channel 32-bit float packed to planar image copy.

- `NppStatus nppiCopy_8u_P3C3R` (const `Npp8u` *const aSrc[3], int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three-channel 8-bit unsigned planar to packed image copy.

- `NppStatus nppiCopy_8u_P4C4R` (const `Npp8u` *const aSrc[4], int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four-channel 8-bit unsigned planar to packed image copy.

- `NppStatus nppiCopy_16u_P3C3R` (const `Npp16u` *const aSrc[3], int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three-channel 16-bit unsigned planar to packed image copy.

- `NppStatus nppiCopy_16u_P4C4R` (const `Npp16u` *const aSrc[4], int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four-channel 16-bit unsigned planar to packed image copy.

- `NppStatus nppiCopy_16s_P3C3R` (const `Npp16s` *const aSrc[3], int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three-channel 16-bit signed planar to packed image copy.

- `NppStatus nppiCopy_16s_P4C4R` (const `Npp16s` *const aSrc[4], int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four-channel 16-bit signed planar to packed image copy.

- `NppStatus nppiCopy_32s_P3C3R` (const `Npp32s` *const aSrc[3], int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three-channel 32-bit signed planar to packed image copy.

- `NppStatus nppiCopy_32s_P4C4R` (const `Npp32s` *const aSrc[4], int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel 32-bit signed planar to packed image copy.
- `NppStatus nppiCopy_32f_P3C3R` (const `Npp32f` *const aSrc[3], int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three-channel 32-bit float planar to packed image copy.
- `NppStatus nppiCopy_32f_P4C4R` (const `Npp32f` *const aSrc[4], int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four-channel 32-bit float planar to packed image copy.

Bit-Depth Conversion

Convert bit-depth up and down.

The integer conversion methods do not involve any scaling. Conversions that reduce bit-depth saturate values exceeding the reduced range to the range's maximum/minimum value. When converting from floating-point values to integer values, a rounding mode can be specified. After rounding to integer values the values get saturated to the destination data type's range.

- `NppStatus nppiConvert_8u16u_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
8-bit unsigned to 16-bit unsigned conversion.
- `NppStatus nppiConvert_16u8u_C1R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
16-bit unsigned to 8-bit unsigned conversion.
- `NppStatus nppiConvert_8u16u_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 8-bit unsigned to 16-bit unsigned conversion.
- `NppStatus nppiConvert_16u8u_C4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 16-bit unsigned to 8-bit unsigned conversion.
- `NppStatus nppiConvert_8u16u_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 8-bit unsigned to 16-bit unsigned conversion, not affecting Alpha.
- `NppStatus nppiConvert_16u8u_AC4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
4 channel 16-bit unsigned to 8-bit unsigned conversion, not affecting Alpha.
- `NppStatus nppiConvert_8u16s_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)
8-bit unsigned to 16-bit signed conversion.

- **NppStatus nppiConvert_16s8u_C1R** (const **Npp16s** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 16-bit signed to 8-bit unsigned conversion.
- **NppStatus nppiConvert_8u16s_C4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 8-bit unsigned to 16-bit signed conversion.
- **NppStatus nppiConvert_16s8u_C4R** (const **Npp16s** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 16-bit signed to 8-bit unsigned conversion, not affecting Alpha.
- **NppStatus nppiConvert_8u16s_AC4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 8-bit unsigned to 16-bit signed conversion, not affecting Alpha.
- **NppStatus nppiConvert_16s8u_AC4R** (const **Npp16s** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 16-bit signed to 8-bit unsigned conversion, not affecting Alpha.
- **NppStatus nppiConvert_16s32f_C1R** (const **Npp16s** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
16-bit signed to 32-bit floating point conversion.
- **NppStatus nppiConvert_32f16s_C1R** (const **Npp32f** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** eRoundMode)
32-bit floating point to 16-bit conversion.
- **NppStatus nppiConvert_8u32f_C1R** (const **Npp8u** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
8-bit unsigned to 32-bit floating point conversion.
- **NppStatus nppiConvert_16u32f_C1R** (const **Npp16u** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
16-bit unsigned to 32-bit floating point conversion.
- **NppStatus nppiConvert_32f16u_C1R** (const **Npp32f** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** eRoundMode)
32-bit floating point to 16-bit unsigned conversion.
- **NppStatus nppiConvert_32f8u_C1R** (const **Npp32f** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** eRoundMode)
32-bit floating point to 8-bit unsigned conversion.
- **NppStatus nppiConvert_16u32s_C1R** (const **Npp16u** *pSrc, int nSrcStep, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
16-bit unsigned to 32-bit signed conversion.
- **NppStatus nppiConvert_16s32s_C1R** (const **Npp16s** *pSrc, int nSrcStep, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
16-bit to 32-bit conversion.

Copy Const Border

Methods for copying images and padding borders with a constant, user-specifiable color.

- `NppStatus nppiCopyConstBorder_8u_C1R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, `Npp8u` nValue)

8-bit unsigned image copy width constant border color.

- `NppStatus nppiCopyConstBorder_8u_C4R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const `Npp8u` aValue[4])

4channel 8-bit unsigned image copy with constant border color.

- `NppStatus nppiCopyConstBorder_8u_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp8u` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const `Npp8u` aValue[3])

4 channel 8-bit unsigned image copy with constant border color.

- `NppStatus nppiCopyConstBorder_32s_C1R` (const `Npp32s` *pSrc, int nSrcStep, `NppiSize` oSrcSizeROI, `Npp32s` *pDst, int nDstStep, `NppiSize` oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, `Npp32s` nValue)

32-bit image copy with constant border color.

Image Transpose

Methods for transposing images of various types.

Like matrix transpose, image transpose is a mirror along the image's diagonal (upper-left to lower-right corner).

- `NppStatus nppiTranspose_8u_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oROI)

8-bit image transpose.

Image Color Channel Swap

Methods for exchanging the color channels of an image.

The methods support arbitrary permutations of the original channels, including replication.

- `NppStatus nppiSwapChannels_8u_C4IR` (`Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, const int aDstOrder[4])

4 channel 8-bit unsigned swap channels, in-place.

7.6.1 Detailed Description

Primitives for initialization, copying and converting image data.

7.6.2 Function Documentation

7.6.2.1 NppStatus nppiConvert_16s32f_C1R (const Npp16s * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

16-bit singedto 32-bit floating point conversion.

For detailed documentation see [nppiConverte_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.2 NppStatus nppiConvert_16s32s_C1R (const Npp16s * *pSrc*, int *nSrcStep*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

16-bit to 32-bit conversion.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.3 NppStatus nppiConvert_16s8u_AC4R (const Npp16s * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 16-bit signed to 8-bit unsigned conversion, not affecting Alpha.

For detailed documentation see [nppiConverte_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.4 NppStatus nppiConvert_16s8u_C1R (const Npp16s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit signed to 8-bit unsigned conversion.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.5 NppStatus nppiConvert_16s8u_C4R (const Npp16s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit signed to 8-bit unsigned conversion, not affecting Alpha.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.6 NppStatus nppiConvert_16u32f_C1R (const Npp16u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

16-bit unsigned to 32-bit floating point conversion.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.7 NppStatus nppiConvert_16u32s_C1R (const Npp16u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

16-bit unsigned to 32-bit signed conversion.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.8 NppStatus nppiConvert_16u8u_AC4R (const Npp16u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit unsigned to 8-bit unsigned conversion, not affecting Alpha.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.9 NppStatus nppiConvert_16u8u_C1R (const Npp16u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

16-bit unsigned to 8-bit unsigned conversion.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.10 NppStatus nppiConvert_16u8u_C4R (const Npp16u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 16-bit unsigned to 8-bit unsigned conversion.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.11 NppStatus nppiConvert_32f16s_C1R (const Npp32f * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, NppRoundMode *eRoundMode*)

32-bit floating point to 16-bit conversion.

For detailed documentation see [nppiConverte_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.12 NppStatus nppiConvert_32f16u_C1R (const Npp32f * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

32-bit floating point to 16-bit unsigned conversion.

For detailed documentation see `nppiConverte_8u16u_C1R()`.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.13 NppStatus nppiConvert_32f8u_C1R (const Npp32f * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode eRoundMode)

32-bit floating point to 8-bit unsigned conversion.

For detailed documentation see `nppiConverte_8u16u_C1R()`.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

eRoundMode Flag specifying how fractional float values are rounded to integer values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.14 NppStatus nppiConvert_8u16s_AC4R (const Npp8u * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 8-bit unsigned to 16-bit signed conversion, not affecting Alpha.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.15 NppStatus nppiConvert_8u16s_C1R (const Npp8u * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit unsigned to 16-bit signed conversion.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.16 NppStatus nppiConvert_8u16s_C4R (const Npp8u * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 8-bit unsigned to 16-bit signed conversion.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.17 NppStatus nppiConvert_8u16u_AC4R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 8-bit unsigned to 16-bit unsigned conversion, not affecting Alpha.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.18 NppStatus nppiConvert_8u16u_C1R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

8-bit unsigned to 16-bit unsigned conversion.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.19 NppStatus nppiConvert_8u16u_C4R (const Npp8u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 8-bit unsigned to 16-bit unsigned conversion.

For detailed documentation see [nppiConvert_8u16u_C1R\(\)](#).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.20 NppStatus nppiConvert_8u32f_C1R (const Npp8u * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

8-bit unsigned to 32-bit floating point conversion.

For detailed documentation see `nppiConverte_8u16u_C1R()`.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.21 NppStatus nppiCopy_16s_AC4MR (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked Operation four channel 16-bit signed image copy, ignoring alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.22 NppStatus nppiCopy_16s_AC4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit image copy, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.23 NppStatus nppiCopy_16s_C1C3R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to three-channel 16-bit signed image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.24 NppStatus nppiCopy_16s_C1C4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Single-channel to four-channel 16-bit signed image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.25 `NppStatus nppiCopy_16s_C1MR (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

[Masked Operation](#) 16-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.26 `NppStatus nppiCopy_16s_C1R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)`

16-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.27 `NppStatus nppiCopy_16s_C3C1R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)`

Three-channel to single-channel 16-bit signed image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.28 NppStatus nppiCopy_16s_C3CR (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Select-channel 16-bit signed image copy for three-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.29 NppStatus nppiCopy_16s_C3MR (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked Operation three channel 16-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.30 NppStatus nppiCopy_16s_C3P3R (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *aDst*[3], int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 16-bit signed packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Planar Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.31 **NppStatus nppiCopy_16s_C4C1R** (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel to single-channel 16-bit signed image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.32 **NppStatus nppiCopy_16s_C4CR** (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Select-channel 16-bit signed image copy for four-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.33 **NppStatus nppiCopy_16s_C4MR** (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked Operation four channel 16-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.34 NppStatus nppiCopy_16s_C4P4R (const Npp16s * pSrc, int nSrcStep, Npp16s *const aDst[4], int nDstStep, NppiSize oSizeROI)

Four-channel 16-bit signed packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Planar Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.35 NppStatus nppiCopy_16s_C4R (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.36 NppStatus nppiCopy_16s_P3C3R (const Npp16s *const aSrc[3], int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel 16-bit signed planar to packed image copy.

Parameters:

aSrc Planar Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.37 NppStatus nppiCopy_16s_P4C4R (const Npp16s *const aSrc[4], int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 16-bit signed planar to packed image copy.

Parameters:

aSrc Planar [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.38 NppStatus nppiCopy_16sc_AC4R (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 16-bit complex image copy, ignoring alpha.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.39 NppStatus nppiCopy_16sc_C1R (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

16-bit complex image copy.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.40 NppStatus nppiCopy_16sc_C2R (const Npp16sc * *pSrc*, int *nSrcStep*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Two-channel 16-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.41 NppStatus nppiCopy_16sc_C3R (const Npp16sc * *pSrc*, int *nSrcStep*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 16-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.42 NppStatus nppiCopy_16sc_C4R (const Npp16sc * *pSrc*, int *nSrcStep*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel 16-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.43 `NppStatus nppiCopy_16u_AC4MR (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

[Masked Operation](#) four channel 16-bit unsigned image copy, ignoring alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.44 `NppStatus nppiCopy_16u_AC4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 16-bit unsigned image copy, not affecting Alpha channel.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.45 `NppStatus nppiCopy_16u_C1C3R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Single-channel to three-channel 16-bit unsigned image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.46 NppStatus nppiCopy_16u_C1C4R (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Single-channel to four-channel 16-bit unsigned image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.47 NppStatus nppiCopy_16u_C1MR (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked Operation 16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.48 NppStatus nppiCopy_16u_C1R (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.49 **NppStatus nppiCopy_16u_C3C1R** (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three-channel to single-channel 16-bit unsigned image copy.

Parameters:

pSrc [Select-Channel Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.50 **NppStatus nppiCopy_16u_C3CR** (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Select-channel 16-bit unsigned image copy for three-channel images.

Parameters:

pSrc [Select-Channel Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Select-Channel Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.51 **NppStatus nppiCopy_16u_C3MR** (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

[Masked Operation](#) three channel 16-bit unsigned image copy.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
pMask [Mask-Image Pointer](#).
nMaskStep [Mask-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.52 NppStatus nppiCopy_16u_C3P3R (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u *const *aDst*[3], int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 16-bit unsigned packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Planar Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.53 NppStatus nppiCopy_16u_C4C1R (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel to single-channel 16-bit unsigned image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.54 NppStatus nppiCopy_16u_C4CR (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Select-channel 16-bit unsigned image copy for four-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.55 `NppStatus nppiCopy_16u_C4MR (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

[Masked Operation](#) four channel 16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.56 `NppStatus nppiCopy_16u_C4P4R (const Npp16u * pSrc, int nSrcStep, Npp16u * const aDst[4], int nDstStep, NppiSize oSizeROI)`

Four-channel 16-bit unsigned packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Planar Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.57 `NppStatus nppiCopy_16u_C4R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 16-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.58 NppStatus nppiCopy_16u_P3C3R (const Npp16u *const aSrc[3], int nSrcStep, Npp16u *pDst, int nDstStep, NppiSize oSizeROI)

Three-channel 16-bit unsigned planar to packed image copy.

Parameters:

aSrc Planar [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.59 NppStatus nppiCopy_16u_P4C4R (const Npp16u *const aSrc[4], int nSrcStep, Npp16u *pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 16-bit unsigned planar to packed image copy.

Parameters:

aSrc Planar [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.60 NppStatus nppiCopy_32f_AC4MR (const Npp32f *pSrc, int nSrcStep, Npp32f *pDst, int nDstStep, NppiSize oSizeROI, const Npp8u *pMask, int nMaskStep)

[Masked Operation](#) four channel 32-bit float image copy, ignoring alpha.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
pMask [Mask-Image Pointer](#).
nMaskStep [Mask-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.61 NppStatus nppiCopy_32f_AC4R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 32-bit floating point image copy, not affecting Alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.62 NppStatus nppiCopy_32f_C1C3R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Single-channel to three-channel 32-bit float image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.63 NppStatus nppiCopy_32f_C1C4R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Single-channel to four-channel 32-bit float image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.64 `NppStatus nppiCopy_32f_C1MR (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked Operation 32-bit float image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.65 `NppStatus nppiCopy_32f_C1R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

32-bit floating point image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.66 `NppStatus nppiCopy_32f_C3C1R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Three-channel to single-channel 32-bit float image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.67 **NppStatus nppiCopy_32f_C3CR** (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Select-channel 32-bit float image copy for three-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.68 **NppStatus nppiCopy_32f_C3MR** (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked Operation three channel 32-bit float image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.69 **NppStatus nppiCopy_32f_C3P3R** (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *aDst*[3], int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 32-bit float packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Planar Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.70 NppStatus nppiCopy_32f_C4C1R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel to single-channel 32-bit float image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.71 NppStatus nppiCopy_32f_C4CR (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Select-channel 32-bit float image copy for four-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.72 NppStatus nppiCopy_32f_C4MR (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked Operation four channel 32-bit float image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.73 **NppStatus nppiCopy_32f_C4P4R** (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f *const *aDst*[4], int *nDstStep*, NppiSize *oSizeROI*)

Four-channel 32-bit float packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Planar Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.74 **NppStatus nppiCopy_32f_C4R** (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 32-bit floating point image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.75 **NppStatus nppiCopy_32f_P3C3R** (const Npp32f *const *aSrc*[3], int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 32-bit float planar to packed image copy.

Parameters:

aSrc Planar Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.76 NppStatus nppiCopy_32f_P4C4R (const Npp32f *const aSrc[4], int nSrcStep, Npp32f *pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 32-bit float planar to packed image copy.

Parameters:

aSrc Planar [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.77 NppStatus nppiCopy_32fc_AC4R (const Npp32fc *pSrc, int nSrcStep, Npp32fc *pDst, int nDstStep, NppiSize oSizeROI)

Four-channel 32-bit floating-point complex image copy, ignoring alpha.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.78 NppStatus nppiCopy_32fc_C1R (const Npp32fc *pSrc, int nSrcStep, Npp32fc *pDst, int nDstStep, NppiSize oSizeROI)

32-bit floating-point complex image copy.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.79 NppStatus nppiCopy_32fc_C2R (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Two-channel 32-bit floating-point complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.80 NppStatus nppiCopy_32fc_C3R (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 32-bit floating-point complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.81 NppStatus nppiCopy_32fc_C4R (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel 32-bit floating-point complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.82 `NppStatus nppiCopy_32s_AC4MR (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

[Masked Operation](#) four channel 32-bit signed image copy, ignoring alpha.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
pMask [Mask-Image Pointer](#).
nMaskStep [Mask-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.83 `NppStatus nppiCopy_32s_AC4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 32-bit image copy, not affecting Alpha.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.84 `NppStatus nppiCopy_32s_C1C3R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Single-channel to three-channel 32-bit signed image copy.

Parameters:

pSrc [Select-Channel Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.85 **NppStatus nppiCopy_32s_C1C4R** (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Single-channel to four-channel 32-bit signed image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.86 **NppStatus nppiCopy_32s_C1MR** (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked Operation 32-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.87 **NppStatus nppiCopy_32s_C1R** (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

32-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.88 NppStatus nppiCopy_32s_C3C1R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Three-channel to single-channel 32-bit signed image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.89 NppStatus nppiCopy_32s_C3CR (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Select-channel 32-bit signed image copy for three-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.90 NppStatus nppiCopy_32s_C3MR (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)

Masked Operation three channel 32-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.91 NppStatus nppiCopy_32s_C3P3R (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s *const *aDst*[3], int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 32-bit signed packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Planar Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.92 NppStatus nppiCopy_32s_C4C1R (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel to single-channel 32-bit signed image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.93 NppStatus nppiCopy_32s_C4CR (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Select-channel 32-bit signed image copy for four-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.94 `NppStatus nppiCopy_32s_C4MR (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

[Masked Operation](#) four channel 32-bit signed image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.95 `NppStatus nppiCopy_32s_C4P4R (const Npp32s * pSrc, int nSrcStep, Npp32s * const aDst[4], int nDstStep, NppiSize oSizeROI)`

Four-channel 32-bit signed packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Planar Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.96 `NppStatus nppiCopy_32s_C4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 32-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.97 **NppStatus nppiCopy_32s_P3C3R** (const Npp32s *const *aSrc*[3], int *nSrcStep*, Npp32s **pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 32-bit signed planar to packed image copy.

Parameters:

aSrc Planar [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.98 **NppStatus nppiCopy_32s_P4C4R** (const Npp32s *const *aSrc*[4], int *nSrcStep*, Npp32s **pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel 32-bit signed planar to packed image copy.

Parameters:

aSrc Planar [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.99 **NppStatus nppiCopy_32sc_AC4R** (const Npp32sc **pSrc*, int *nSrcStep*, Npp32sc **pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel 32-bit complex image copy, ignoring alpha.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.100 NppStatus nppiCopy_32sc_C1R (const Npp32sc * *pSrc*, int *nSrcStep*, Npp32sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

32-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.101 NppStatus nppiCopy_32sc_C2R (const Npp32sc * *pSrc*, int *nSrcStep*, Npp32sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Two-channel 32-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.102 NppStatus nppiCopy_32sc_C3R (const Npp32sc * *pSrc*, int *nSrcStep*, Npp32sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 32-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.103 NppStatus nppiCopy_32sc_C4R (const Npp32sc * *pSrc*, int *nSrcStep*, Npp32sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel 32-bit complex image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.104 NppStatus nppiCopy_8s_AC4R (const Npp8s * *pSrc*, int *nSrcStep*, Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel 8-bit image copy, ignoring alpha channel.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.105 NppStatus nppiCopy_8s_C1R (const Npp8s * *pSrc*, int *nSrcStep*, Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.106 NppStatus nppiCopy_8s_C2R (const Npp8s * *pSrc*, int *nSrcStep*, Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Two-channel 8-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.107 NppStatus nppiCopy_8s_C3R (const Npp8s * *pSrc*, int *nSrcStep*, Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 8-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.108 NppStatus nppiCopy_8s_C4R (const Npp8s * *pSrc*, int *nSrcStep*, Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel 8-bit image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.109 `NppStatus nppiCopy_8u_AC4MR (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

[Masked Operation](#) four channel 8-bit unsigned image copy, ignoring alpha.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
pMask [Mask-Image Pointer](#).
nMaskStep [Mask-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.110 `NppStatus nppiCopy_8u_AC4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 8-bit unsigned image copy, not affecting Alpha channel.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.111 `NppStatus nppiCopy_8u_C1C3R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Single-channel to three-channel 8-bit unsigned image copy.

Parameters:

pSrc [Select-Channel Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.112 NppStatus nppiCopy_8u_C1C4R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Single-channel to four-channel 8-bit unsigned image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.113 NppStatus nppiCopy_8u_C1MR (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked Operation 8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.114 NppStatus nppiCopy_8u_C1R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.115 `NppStatus nppiCopy_8u_C3C1R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three-channel to single-channel 8-bit unsigned image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.116 `NppStatus nppiCopy_8u_C3CR (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Select-channel 8-bit unsigned image copy for three-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.117 `NppStatus nppiCopy_8u_C3MR (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked Operation three channel 8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.118 NppStatus nppiCopy_8u_C3P3R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u *const *aDst*[3], int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 8-bit unsigned packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Planar Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.119 NppStatus nppiCopy_8u_C4C1R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel to single-channel 8-bit unsigned image copy.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.120 NppStatus nppiCopy_8u_C4CR (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Select-channel 8-bit unsigned image copy for four-channel images.

Parameters:

pSrc Select-Channel Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Select-Channel Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.121 `NppStatus nppiCopy_8u_C4MR (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

[Masked Operation](#) four channel 8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.122 `NppStatus nppiCopy_8u_C4P4R (const Npp8u * pSrc, int nSrcStep, Npp8u * const aDst[4], int nDstStep, NppiSize oSizeROI)`

Four-channel 8-bit unsigned packed to planar image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
aDst Planar Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.123 `NppStatus nppiCopy_8u_C4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 8-bit unsigned image copy.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.124 NppStatus nppiCopy_8u_P3C3R (const Npp8u *const *aSrc*[3], int *nSrcStep*, Npp8u **pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three-channel 8-bit unsigned planar to packed image copy.

Parameters:

aSrc Planar [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.125 NppStatus nppiCopy_8u_P4C4R (const Npp8u *const *aSrc*[4], int *nSrcStep*, Npp8u **pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four-channel 8-bit unsigned planar to packed image copy.

Parameters:

aSrc Planar [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.126 NppStatus nppiCopyConstBorder_32s_C1R (const Npp32s **pSrc*, int *nSrcStep*, NppiSize *oSrcSizeROI*, Npp32s **pDst*, int *nDstStep*, NppiSize *oDstSizeROI*, int *nTopBorderHeight*, int *nLeftBorderWidth*, Npp32s *nValue*)

32-bit image copy with constant border color.

See [nppiCopyConstBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSizeROI Size of the source region-of-interest.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

nValue Border luminance value.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.127 `NppStatus nppiCopyConstBorder_8u_AC4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp8u aValue[3])`

4 channel 8-bit unsigned image copy with constant border color.

See [nppiCopyConstBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSizeROI Size of the source region-of-interest.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGB values of the border pixels. Because this method does not affect the destination image's alpha channel, only three components of the border color are needed.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.128 `NppStatus nppiCopyConstBorder_8u_C1R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, Npp8u nValue)`

8-bit unsigned image copy width constant border color.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSizeROI Size of the source region of pixels.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oDstSizeROI Size (width, height) of the destination region, i.e. the region that gets filled with data from the source image (inner part) and constant border color (outer part).

nTopBorderHeight Height (in pixels) of the top border. The height of the border at the bottom of the destination ROI is implicitly defined by the size of the source ROI: $nBottomBorderHeight = oDstSizeROI.height - nTopBorderHeight - oSrcSizeROI.height$.

nLeftBorderWidth Width (in pixels) of the left border. The width of the border at the right side of the destination ROI is implicitly defined by the size of the source ROI: $nRightBorderWidth = oDstSizeROI.width - nLeftBorderWidth - oSrcSizeROI.width$.

nValue The pixel value to be set for border pixels.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.129 `NppStatus nppiCopyConstBorder_8u_C4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSrcSizeROI, Npp8u * pDst, int nDstStep, NppiSize oDstSizeROI, int nTopBorderHeight, int nLeftBorderWidth, const Npp8u aValue[4])`

4channel 8-bit unsigned image copy with constant border color.

See [npapiCopyConstBorder_8u_C1R\(\)](#) for detailed documentation.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSizeROI Size of the source region-of-interest.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oDstSizeROI Size of the destination region-of-interest.

nTopBorderHeight Height of top border.

nLeftBorderWidth Width of left border.

aValue Vector of the RGBA values of the border pixels to be set.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.130 `NppStatus nppiSet_16s_AC4MR (const Npp16s aValues[3], Npp16s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 4 channel 16-bit image set method, not affecting Alpha channel.

For RGBA images, this method allows setting of the RGB values without changing the contents of the alpha-channel (fourth channel).

Parameters:

aValues Three-channel array containing the pixel-value to be set.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.131 NppStatus nppiSet_16s_AC4R (const Npp16s aValues[3], Npp16s *pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit image set method, not affecting Alpha channel.

For RGBA images, this method allows setting of the RGB values without changing the contents of the alpha-channel (fourth channel).

Parameters:

aValues Three-channel array containing the pixel-value to be set.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.132 NppStatus nppiSet_16s_C1MR (Npp16s nValue, Npp16s *pDst, int nDstStep, NppiSize oSizeROI, const Npp8u *pMask, int nMaskStep)

Masked 16-bit image set.

Parameters:

nValue New pixel value.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.133 NppStatus nppiSet_16s_C1R (Npp16s nValue, Npp16s *pDst, int nDstStep, NppiSize oSizeROI)

16-bit image set.

Parameters:

nValue New pixel value.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.134 NppStatus nppiSet_16s_C2R (const Npp16s aValues[2], Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

2 channel 16-bit image set.

Parameters:

aValues New pixel value.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.135 NppStatus nppiSet_16s_C4CR (Npp16s nValue, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)

4 channel 16-bit unsigned image set affecting only single channel.

For RGBA images, this method allows setting of a single of the four (RGBA) values without changing the contents of the other three channels. The channel is selected via the *pDst* pointer. The pointer needs to point to the actual first value to be set, e.g. in order to set the R-channel (first channel), one would pass *pDst* unmodified, since its value actually points to the r channel. If one wanted to modify the B channel (second channel), one would pass *pDst* + 2 to the function.

Parameters:

nValue The pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.136 NppStatus nppiSet_16s_C4MR (const Npp16s *aValues*[4], Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 4 channel 16-bit image set.

Parameters:

aValues New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.137 NppStatus nppiSet_16s_C4R (const Npp16s *aValues*[4], Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 16-bit image set.

Parameters:

aValues New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.138 NppStatus nppiSet_16sc_AC4R (Npp16sc *aValue*[3], Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

16-bit complex integer four-channel image set ignoring alpha.

Parameters:

aValue New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.139 NppStatus nppiSet_16sc_C1R (Npp16sc oValue, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

16-bit complex integer image set.

Parameters:

oValue New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.140 NppStatus nppiSet_16sc_C2R (Npp16sc aValue[2], Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

16-bit complex integer two-channel image set.

Parameters:

aValue New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.141 NppStatus nppiSet_16sc_C3R (Npp16sc aValue[3], Npp16sc * pDst, int nDstStep, NppiSize oSizeROI)

16-bit complex integer three-channel image set.

Parameters:

aValue New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.142 **NppStatus nppiSet_16sc_C4R** (Npp16sc *aValue*[4], Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

16-bit complex integer four-channel image set.

Parameters:

aValue New pixel value.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.143 **NppStatus nppiSet_16u_AC4MR** (const Npp16u *aValues*[3], Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 4 channel 16-bit unsigned image set method, not affecting Alpha channel.

For RGBA images, this method allows setting of the RGB values without changing the contents of the alpha-channel (fourth channel).

Parameters:

aValues Three-channel array containing the pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.
nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.144 **NppStatus nppiSet_16u_AC4R** (const Npp16u *aValues*[3], Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 16-bit unsigned image set method, not affecting Alpha channel.

For RGBA images, this method allows setting of the RGB values without changing the contents of the alpha-channel (fourth channel).

Parameters:

aValues Three-channel array containing the pixel-value to be set.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.145 NppStatus nppiSet_16u_C1MR (Npp16u *nValue*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 16-bit unsigned image set.

Parameters:

nValue New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.146 NppStatus nppiSet_16u_C1R (Npp16u *nValue*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

16-bit unsigned image set.

Parameters:

nValue New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.147 NppStatus nppiSet_16u_C2R (const Npp16u *aValues*[2], Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

2 channel 16-bit unsigned image set.

Parameters:

aValues New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.148 NppStatus nppiSet_16u_C4CR (Npp16u *nValue*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 16-bit unsigned image set affecting only single channel.

For RGBA images, this method allows setting of a single of the four (RGBA) values without changing the contents of the other three channels. The channel is selected via the *pDst* pointer. The pointer needs to point to the actual first value to be set, e.g. in order to set the R-channel (first channel), one would pass *pDst* unmodified, since its value actually points to the r channel. If one wanted to modify the B channel (second channel), one would pass *pDst* + 2 to the function.

Parameters:

nValue The pixel-value to be set.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.149 NppStatus nppiSet_16u_C4MR (const Npp16u *aValues*[4], Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 4 channel 16-bit unsigned image set.

Parameters:

aValues New pixel value.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.150 NppStatus nppiSet_16u_C4R (const Npp16u *aValues*[4], Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 16-bit unsigned image set.

Parameters:

aValues New pixel value.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.151 `NppStatus nppiSet_32f_AC4MR (const Npp32f aValues[3], Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 4 channel 32-bit floating point image set method, not affecting Alpha channel.

For RGBA images, this method allows setting of the RGB values without changing the contents of the alpha-channel (fourth channel).

Parameters:

aValues Three-channel array containing the pixel-value to be set.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.152 `NppStatus nppiSet_32f_AC4R (const Npp32f aValues[3], Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 32-bit floating point image set method, not affecting Alpha channel.

For RGBA images, this method allows setting of the RGB values without changing the contents of the alpha-channel (fourth channel).

Parameters:

aValues Three-channel array containing the pixel-value to be set.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.153 **NppStatus nppiSet_32f_C1MR** (Npp32f *nValue*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 32-bit floating point image set.

Parameters:

nValue New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.154 **NppStatus nppiSet_32f_C1R** (Npp32f *nValue*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

32-bit floating point image set.

Parameters:

nValue New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.155 **NppStatus nppiSet_32f_C4CR** (Npp32f *nValue*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 32-bit floating point image set affecting only single channel.

For RGBA images, this method allows setting of a single of the four (RGBA) values without changing the contents of the other three channels. The channel is selected via the *pDst* pointer. The pointer needs to point to the actual first value to be set, e.g. in order to set the R-channel (first channel), one would pass *pDst* unmodified, since its value actually points to the r channel. If one wanted to modify the B channel (second channel), one would pass *pDst* + 2 to the function.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.156 `NppStatus nppiSet_32f_C4MR (const Npp32f aValues[4], Npp32f * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 4 channel 32-bit floating point image set.

Parameters:

aValues New pixel value.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.157 `NppStatus nppiSet_32f_C4R (const Npp32f aValues[4], Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 32-bit floating point image set.

Parameters:

aValues New pixel value.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.158 `NppStatus nppiSet_32fc_AC4R (Npp32fc aValue[3], Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

32-bit complex four-channel image set ignoring alpha.

Parameters:

aValue New pixel value.

pDst [Destination-Image Pointer](#).

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.159 `NppStatus nppiSet_32fc_C1R (Npp32fc oValue, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Single channel 32-bit complex image set.

Parameters:

oValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.160 `NppStatus nppiSet_32fc_C2R (Npp32fc aValue[2], Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Two channel 32-bit complex image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.161 `NppStatus nppiSet_32fc_C3R (Npp32fc aValue[3], Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Three channel 32-bit complex image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.162 `NppStatus nppiSet_32fc_C4R (Npp32fc aValue[4], Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four channel 32-bit complex image set.

Parameters:

aValue The pixel-value to be set.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.163 `NppStatus nppiSet_32s_AC4MR (const Npp32s aValues[3], Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 4 channel 16-bit image set method, not affecting Alpha channel.

For RGBA images, this method allows setting of the RGB values without changing the contents of the alpha-channel (fourth channel).

Parameters:

aValues Three-channel array containing the pixel-value to be set.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.164 `NppStatus nppiSet_32s_AC4R (const Npp32s aValues[3], Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 16-bit image set method, not affecting Alpha channel.

For RGBA images, this method allows setting of the RGB values without changing the contents of the alpha-channel (fourth channel).

Parameters:

aValues Three-channel array containing the pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.165 `NppStatus nppiSet_32s_C1MR (Npp32s nValue, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 32-bit image set.

Parameters:

nValue New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.166 `NppStatus nppiSet_32s_C1R (Npp32s nValue, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

32-bit image set.

Parameters:

nValue New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.167 NppStatus nppiSet_32s_C4CR (Npp32s *nValue*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 32-bit unsigned image set affecting only single channel.

For RGBA images, this method allows setting of a single of the four (RGBA) values without changing the contents of the other three channels. The channel is selected via the *pDst* pointer. The pointer needs to point to the actual first value to be set, e.g. in order to set the R-channel (first channel), one would pass *pDst* unmodified, since its value actually points to the r channel. If one wanted to modify the B channel (second channel), one would pass *pDst* + 2 to the function.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.168 NppStatus nppiSet_32s_C4MR (const Npp32s *aValues*[4], Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 4 channel 32-bit image set.

Parameters:

aValues New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.169 NppStatus nppiSet_32s_C4R (const Npp32s *aValues*[4], Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 32-bit image set.

Parameters:

aValues New pixel value.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.170 `NppStatus nppiSet_32sc_AC4R (Npp32sc aValue[3], Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)`

32-bit complex integer four-channel image set ignoring alpha.

Parameters:

aValue New pixel value.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.171 `NppStatus nppiSet_32sc_C1R (Npp32sc oValue, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)`

Single channel 32-bit complex integer image set.

Parameters:

oValue The pixel-value to be set.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.172 `NppStatus nppiSet_32sc_C2R (Npp32sc aValue[2], Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)`

Two channel 32-bit complex integer image set.

Parameters:

aValue The pixel-value to be set.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.173 NppStatus nppiSet_32sc_C3R (Npp32sc aValue[3], Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)

Three channel 32-bit complex integer image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.174 NppStatus nppiSet_32sc_C4R (Npp32sc aValue[4], Npp32sc * pDst, int nDstStep, NppiSize oSizeROI)

Four channel 32-bit complex integer image set.

Parameters:

aValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.175 NppStatus nppiSet_8s_AC4R (Npp8s aValue[3], Npp8s * pDst, int nDstStep, NppiSize oSizeROI)

8-bit four-channel image set ignoring alpha channel.

Parameters:

aValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.176 NppStatus nppiSet_8s_C1R (Npp8s *nValue*, Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit image set.

Parameters:

nValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.177 NppStatus nppiSet_8s_C2R (Npp8s *aValue*[2], Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit two-channel image set.

Parameters:

aValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.178 NppStatus nppiSet_8s_C3R (Npp8s *aValue*[3], Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit three-channel image set.

Parameters:

aValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.179 NppStatus nppiSet_8s_C4R (Npp8s *aValue*[4], Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

8-bit four-channel image set.

Parameters:

aValue The pixel value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.180 NppStatus nppiSet_8u_AC4MR (const Npp8u *aValues*[3], Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, int *nMaskStep*)

Masked 4 channel 8-bit unsigned image set method, not affecting Alpha channel.

For RGBA images, this method allows setting of the RGB values without changing the contents of the alpha-channel (fourth channel).

Parameters:

aValues Three-channel array containing the pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.181 NppStatus nppiSet_8u_AC4R (const Npp8u *aValues*[3], Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 8-bit unsigned image set method, not affecting Alpha channel.

For RGBA images, this method allows setting of the RGB values without changing the contents of the alpha-channel (fourth channel).

Parameters:

aValues Three-channel array containing the pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.182 `NppStatus nppiSet_8u_C1MR (Npp8u nValue, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 8-bit unsigned image set.

The 8-bit mask image affects setting of the respective pixels in the destination image. If the mask value is zero (0) the pixel is not set, if the mask is non-zero, the corresponding destination pixel is set to specified value.

Parameters:

nValue The pixel value to be set.

pDst Pointer [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.183 `NppStatus nppiSet_8u_C1R (Npp8u nValue, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

8-bit unsigned image set.

Parameters:

nValue The pixel value to be set.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.184 `NppStatus nppiSet_8u_C4CR (Npp8u nValue, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 8-bit unsigned image set affecting only single channel.

For RGBA images, this method allows setting of a single of the four (RGBA) values without changing the contents of the other three channels. The channel is selected via the *pDst* pointer. The pointer needs to point to the actual first value to be set, e.g. in order to set the R-channel (first channel), one would pass *pDst* unmodified, since its value actually points to the r channel. If one wanted to modify the B channel (second channel), one would pass *pDst* + 2 to the function.

Parameters:

nValue The pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.185 `NppStatus nppiSet_8u_C4MR (const Npp8u aValues[4], Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, int nMaskStep)`

Masked 4 channel 8-bit unsigned image set.

Parameters:

aValues Four-channel array containing the pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pMask Pointer to the mask image. This is a single channel 8-bit unsigned int image.

nMaskStep Number of bytes between line starts of successive lines in the mask image.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.186 `NppStatus nppiSet_8u_C4R (const Npp8u aValues[4], Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 8-bit unsigned image set.

Parameters:

aValues Four-channel array containing the pixel-value to be set.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.6.2.187 `NppStatus nppiSwapChannels_8u_C4IR (Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, const int aDstOrder[4])`

4 channel 8-bit unsigned swap channels, in-place.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aDstOrder Integer array describing how channel values are permuted. The n-th entry of the array contains the number of the channel that is stored in the n-th channel of the output image. E.g. Given an RGBA image, aDstOrder = [3,2,1,0] converts this to ABGR channel order.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.6.2.188 `NppStatus npptTranspose_8u_C1R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oROI)`

8-bit image transpose.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst Pointer to the destination ROI.

nDstStep [Destination-Image Line Step](#).

oROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7 Arithmetic and Logical Operations

AddC

Adds a constant value to each pixel of an image.

- **NppStatus nppiAddC_8u_C1RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** nConstant, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel image add constant, scale, then clamp to saturated value.
- **NppStatus nppiAddC_8u_C1IRSfs** (const **Npp8u** nConstant, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel in place image add constant, scale, then clamp to saturated value.
- **NppStatus nppiAddC_8u_C3RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel image add constant, scale, then clamp to saturated value.
- **NppStatus nppiAddC_8u_C3IRSfs** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel 8-bit unsigned char in place image add constant, scale, then clamp to saturated value.
- **NppStatus nppiAddC_8u_AC4RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel with unmodified alpha image add constant, scale, then clamp to saturated value.
- **NppStatus nppiAddC_8u_AC4IRSfs** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel with unmodified alpha in place image add constant, scale, then clamp to saturated value.
- **NppStatus nppiAddC_8u_C4RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel image add constant, scale, then clamp to saturated value.
- **NppStatus nppiAddC_8u_C4IRSfs** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel in place image add constant, scale, then clamp to saturated value.
- **NppStatus nppiAddC_16u_C1RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** nConstant, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel image add constant, scale, then clamp to saturated value.
- **NppStatus nppiAddC_16u_C1IRSfs** (const **Npp16u** nConstant, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel in place image add constant, scale, then clamp to saturated value.
- **NppStatus nppiAddC_16u_C3RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pConstants, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 16-bit unsigned short channel image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16u_C3IRSfs` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit unsigned short channel in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16u_AC4RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16u_AC4IRSfs` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16u_C4RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16u_C4IRSfs` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16s_C1RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` nConstant, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short channel image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16s_C1IRSfs` (const `Npp16s` nConstant, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short channel in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16s_C3RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short channel image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16s_C3IRSfs` (const `Npp16s` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short channel in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16s_AC4RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel with unmodified alpha image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16s_AC4IRSfs` (const `Npp16s` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel with unmodified alpha in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16s_C4RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16s_C4IRSfs` (const `Npp16s` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16sc_C1RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` nConstant, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16sc_C1RSfs` (const `Npp16sc` nConstant, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16sc_C3RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` *pConstants, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16sc_C3RSfs` (const `Npp16sc` *pConstants, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16sc_AC4RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` *pConstants, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_16sc_AC4RSfs` (const `Npp16sc` *pConstants, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_32s_C1RSfs` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` nConstant, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 32-bit signed integer channel image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_32s_C1RSfs` (const `Npp32s` nConstant, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 32-bit signed integer channel in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_32s_C3RSfs` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 32-bit signed integer channel image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_32s_C3RSfs` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 32-bit signed integer channel in place image add constant, scale, then clamp to saturated value.

- `NppStatus nppiAddC_32sc_C1RSfs` (const `Npp32sc` *pSrc1, int nSrc1Step, const `Npp32sc` nConstant, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image add constant, scale, then clamp to saturated value.
- `NppStatus nppiAddC_32sc_C1IRSfs` (const `Npp32sc` nConstant, `Npp32sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image add constant, scale, then clamp to saturated value.
- `NppStatus nppiAddC_32sc_C3RSfs` (const `Npp32sc` *pSrc1, int nSrc1Step, const `Npp32sc` *pConstants, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image add constant, scale, then clamp to saturated value.
- `NppStatus nppiAddC_32sc_C3IRSfs` (const `Npp32sc` *pConstants, `Npp32sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image add constant, scale, then clamp to saturated value.
- `NppStatus nppiAddC_32sc_AC4RSfs` (const `Npp32sc` *pSrc1, int nSrc1Step, const `Npp32sc` *pConstants, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha image add constant, scale, then clamp to saturated value.
- `NppStatus nppiAddC_32sc_AC4IRSfs` (const `Npp32sc` *pConstants, `Npp32sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image add constant, scale, then clamp to saturated value.
- `NppStatus nppiAddC_32f_C1R` (const `Npp32f` *pSrc1, int nSrc1Step, const `Npp32f` nConstant, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
One 32-bit floating point channel image add constant.
- `NppStatus nppiAddC_32f_C1IR` (const `Npp32f` nConstant, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 32-bit floating point channel in place image add constant.
- `NppStatus nppiAddC_32f_C3R` (const `Npp32f` *pSrc1, int nSrc1Step, const `Npp32f` *pConstants, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 32-bit floating point channel image add constant.
- `NppStatus nppiAddC_32f_C3IR` (const `Npp32f` *pConstants, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 32-bit floating point channel in place image add constant.
- `NppStatus nppiAddC_32f_AC4R` (const `Npp32f` *pSrc1, int nSrc1Step, const `Npp32f` *pConstants, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 32-bit floating point channel with unmodified alpha image add constant.
- `NppStatus nppiAddC_32f_AC4IR` (const `Npp32f` *pConstants, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel with unmodified alpha in place image add constant.

- `NppStatus nppiAddC_32f_C4R` (const `Npp32f` *pSrc1, int nSrc1Step, const `Npp32f` *pConstants, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel image add constant.

- `NppStatus nppiAddC_32f_C4IR` (const `Npp32f` *pConstants, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel in place image add constant.

- `NppStatus nppiAddC_32fc_C1R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` nConstant, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image add constant.

- `NppStatus nppiAddC_32fc_C1IR` (const `Npp32fc` nConstant, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image add constant.

- `NppStatus nppiAddC_32fc_C3R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pConstants, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image add constant.

- `NppStatus nppiAddC_32fc_C3IR` (const `Npp32fc` *pConstants, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image add constant.

- `NppStatus nppiAddC_32fc_AC4R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pConstants, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha image add constant.

- `NppStatus nppiAddC_32fc_AC4IR` (const `Npp32fc` *pConstants, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha in place image add constant.

- `NppStatus nppiAddC_32fc_C4R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pConstants, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image add constant.

- `NppStatus nppiAddC_32fc_C4IR` (const `Npp32fc` *pConstants, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image add constant.

MulC

Multiplies each pixel of an image by a constant value.

- **NppStatus nppiMulC_8u_C1RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** nConstant, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_8u_C1IRSfs** (const **Npp8u** nConstant, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_8u_C3RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_8u_C3IRSfs** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel 8-bit unsigned char in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_8u_AC4RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel with unmodified alpha image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_8u_AC4IRSfs** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel with unmodified alpha in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_8u_C4RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_8u_C4IRSfs** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16u_C1RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** nConstant, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16u_C1IRSfs** (const **Npp16u** nConstant, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16u_C3RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pConstants, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel image multiply by constant, scale, then clamp to saturated value.

- **NppStatus nppiMulC_16u_C3IRSfs** (const **Npp16u** *pConstants, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16u_AC4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pConstants, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel with unmodified alpha image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16u_AC4IRSfs** (const **Npp16u** *pConstants, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel with unmodified alpha in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16u_C4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pConstants, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16u_C4IRSfs** (const **Npp16u** *pConstants, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16s_C1RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** nConstant, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16s_C1IRSfs** (const **Npp16s** nConstant, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16s_C3RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pConstants, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short channel image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16s_C3IRSfs** (const **Npp16s** *pConstants, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short channel in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16s_AC4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pConstants, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16s_AC4IRSfs** (const **Npp16s** *pConstants, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_16s_C4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pConstants, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit signed short channel image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_16s_C4IRSfs` (const `Npp16s` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel in place image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_16sc_C1RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` nConstant, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_16sc_C1IRSfs` (const `Npp16sc` nConstant, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_16sc_C3RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` *pConstants, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_16sc_C3IRSfs` (const `Npp16sc` *pConstants, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_16sc_AC4RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` *pConstants, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_16sc_AC4IRSfs` (const `Npp16sc` *pConstants, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_32s_C1RSfs` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` nConstant, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 32-bit signed integer channel image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_32s_C1IRSfs` (const `Npp32s` nConstant, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 32-bit signed integer channel in place image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_32s_C3RSfs` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 32-bit signed integer channel image multiply by constant, scale, then clamp to saturated value.

- `NppStatus nppiMulC_32s_C3IRSfs` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 32-bit signed integer channel in place image multiply by constant, scale, then clamp to saturated value.

- **NppStatus nppiMulC_32sc_C1RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** nConstant, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_32sc_C1IRSfs** (const **Npp32sc** nConstant, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_32sc_C3RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pConstants, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_32sc_C3IRSfs** (const **Npp32sc** *pConstants, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_32sc_AC4RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pConstants, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_32sc_AC4IRSfs** (const **Npp32sc** *pConstants, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image multiply by constant, scale, then clamp to saturated value.
- **NppStatus nppiMulC_32f_C1R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** nConstant, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel image multiply by constant.
- **NppStatus nppiMulC_32f_C1IR** (const **Npp32f** nConstant, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel in place image multiply by constant.
- **NppStatus nppiMulC_32f_C3R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pConstants, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel image multiply by constant.
- **NppStatus nppiMulC_32f_C3IR** (const **Npp32f** *pConstants, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel in place image multiply by constant.
- **NppStatus nppiMulC_32f_AC4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pConstants, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel with unmodified alpha image multiply by constant.

- **NppStatus nppiMulC_32f_AC4IR** (const **Npp32f** *pConstants, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel with unmodified alpha in place image multiply by constant.
- **NppStatus nppiMulC_32f_C4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pConstants, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel image multiply by constant.
- **NppStatus nppiMulC_32f_C4IR** (const **Npp32f** *pConstants, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel in place image multiply by constant.
- **NppStatus nppiMulC_32fc_C1R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** nConstant, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image multiply by constant.
- **NppStatus nppiMulC_32fc_C1IR** (const **Npp32fc** nConstant, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image multiply by constant.
- **NppStatus nppiMulC_32fc_C3R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pConstants, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image multiply by constant.
- **NppStatus nppiMulC_32fc_C3IR** (const **Npp32fc** *pConstants, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image multiply by constant.
- **NppStatus nppiMulC_32fc_AC4R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pConstants, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha image multiply by constant.
- **NppStatus nppiMulC_32fc_AC4IR** (const **Npp32fc** *pConstants, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha in place image multiply by constant.
- **NppStatus nppiMulC_32fc_C4R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pConstants, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image multiply by constant.
- **NppStatus nppiMulC_32fc_C4IR** (const **Npp32fc** *pConstants, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image multiply by constant.

MulCScale

Multiplies each pixel of an image by a constant value then scales the result by the maximum value for the data bit width.

- **NppStatus nppiMulCScale_8u_C1R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** nConstant, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel image multiply by constant and scale by max bit width value.
- **NppStatus nppiMulCScale_8u_C1IR** (const **Npp8u** nConstant, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel in place image multiply by constant and scale by max bit width value.
- **NppStatus nppiMulCScale_8u_C3R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel image multiply by constant and scale by max bit width value.
- **NppStatus nppiMulCScale_8u_C3IR** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel 8-bit unsigned char in place image multiply by constant and scale by max bit width value.
- **NppStatus nppiMulCScale_8u_AC4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 8-bit unsigned char channel with unmodified alpha image multiply by constant and scale by max bit width value.
- **NppStatus nppiMulCScale_8u_AC4IR** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 8-bit unsigned char channel with unmodified alpha in place image multiply by constant, scale and scale by max bit width value.
- **NppStatus nppiMulCScale_8u_C4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 8-bit unsigned char channel image multiply by constant and scale by max bit width value.
- **NppStatus nppiMulCScale_8u_C4IR** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 8-bit unsigned char channel in place image multiply by constant and scale by max bit width value.
- **NppStatus nppiMulCScale_16u_C1R** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** nConstant, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 16-bit unsigned short channel image multiply by constant and scale by max bit width value.
- **NppStatus nppiMulCScale_16u_C1IR** (const **Npp16u** nConstant, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 16-bit unsigned short channel in place image multiply by constant and scale by max bit width value.
- **NppStatus nppiMulCScale_16u_C3R** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pConstants, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 16-bit unsigned short channel image multiply by constant and scale by max bit width value.

- `NppStatus nppiMulCScale_16u_C3IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel in place image multiply by constant and scale by max bit width value.

- `NppStatus nppiMulCScale_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel with unmodified alpha image multiply by constant and scale by max bit width value.

- `NppStatus nppiMulCScale_16u_AC4IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel with unmodified alpha in place image multiply by constant and scale by max bit width value.

- `NppStatus nppiMulCScale_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image multiply by constant and scale by max bit width value.

- `NppStatus nppiMulCScale_16u_C4IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image multiply by constant and scale by max bit width value.

SubC

Subtracts a constant value from each pixel of an image.

- `NppStatus nppiSubC_8u_C1RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` nConstant, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_8u_C1IRSfs` (const `Npp8u` nConstant, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel in place image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_8u_C3RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 8-bit unsigned char channel image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_8u_C3IRSfs` (const `Npp8u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 8-bit unsigned char channel 8-bit unsigned char in place image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_8u_AC4RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel with unmodified alpha image subtract constant, scale, then clamp to saturated value.

- **NppStatus nppiSubC_8u_AC4IRSfs** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel with unmodified alpha in place image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_8u_C4RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_8u_C4IRSfs** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel in place image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_16u_C1RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** nConstant, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_16u_C1IRSfs** (const **Npp16u** nConstant, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel in place image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_16u_C3RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pConstants, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_16u_C3IRSfs** (const **Npp16u** *pConstants, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel in place image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_16u_AC4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pConstants, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel with unmodified alpha image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_16u_AC4IRSfs** (const **Npp16u** *pConstants, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel with unmodified alpha in place image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_16u_C4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pConstants, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_16u_C4IRSfs** (const **Npp16u** *pConstants, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel in place image subtract constant, scale, then clamp to saturated value.
- **NppStatus nppiSubC_16s_C1RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** nConstant, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_16s_C1IRSfs` (const `Npp16s` nConstant, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16s_C3RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Three 16-bit signed short channel image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16s_C3IRSfs` (const `Npp16s` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Three 16-bit signed short channel in place image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16s_AC4RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16s_AC4IRSfs` (const `Npp16s` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha in place image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16s_C4RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Four 16-bit signed short channel image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16s_C4IRSfs` (const `Npp16s` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Four 16-bit signed short channel in place image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16sc_C1RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` nConstant, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16sc_C1IRSfs` (const `Npp16sc` nConstant, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16sc_C3RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` *pConstants, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16sc_C3IRSfs` (const `Npp16sc` *pConstants, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image subtract constant, scale, then clamp to saturated value.
- `NppStatus nppiSubC_16sc_AC4RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` *pConstants, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_16sc_AC4IRSfs` (const `Npp16sc` *pConstants, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32s_C1RSfs` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` nConstant, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 32-bit signed integer channel image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32s_C1IRSfs` (const `Npp32s` nConstant, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 32-bit signed integer channel in place image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32s_C3RSfs` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 32-bit signed integer channel image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32s_C3IRSfs` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 32-bit signed integer channel in place image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32sc_C1RSfs` (const `Npp32sc` *pSrc1, int nSrc1Step, const `Npp32sc` nConstant, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32sc_C1IRSfs` (const `Npp32sc` nConstant, `Npp32sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32sc_C3RSfs` (const `Npp32sc` *pSrc1, int nSrc1Step, const `Npp32sc` *pConstants, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32sc_C3IRSfs` (const `Npp32sc` *pConstants, `Npp32sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32sc_AC4RSfs` (const `Npp32sc` *pSrc1, int nSrc1Step, const `Npp32sc` *pConstants, `Npp32sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32sc_AC4IRSfs` (const `Npp32sc` *pConstants, `Npp32sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image subtract constant, scale, then clamp to saturated value.

- `NppStatus nppiSubC_32f_C1R` (const `Npp32f` *pSrc1, int nSrc1Step, const `Npp32f` nConstant, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 32-bit floating point channel image subtract constant.

- `NppStatus nppiSubC_32f_C1IR` (const `Npp32f` nConstant, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 32-bit floating point channel in place image subtract constant.

- `NppStatus nppiSubC_32f_C3R` (const `Npp32f` *pSrc1, int nSrc1Step, const `Npp32f` *pConstants, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 32-bit floating point channel image subtract constant.

- `NppStatus nppiSubC_32f_C3IR` (const `Npp32f` *pConstants, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 32-bit floating point channel in place image subtract constant.

- `NppStatus nppiSubC_32f_AC4R` (const `Npp32f` *pSrc1, int nSrc1Step, const `Npp32f` *pConstants, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel with unmodified alpha image subtract constant.

- `NppStatus nppiSubC_32f_AC4IR` (const `Npp32f` *pConstants, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel with unmodified alpha in place image subtract constant.

- `NppStatus nppiSubC_32f_C4R` (const `Npp32f` *pSrc1, int nSrc1Step, const `Npp32f` *pConstants, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel image subtract constant.

- `NppStatus nppiSubC_32f_C4IR` (const `Npp32f` *pConstants, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel in place image subtract constant.

- `NppStatus nppiSubC_32fc_C1R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` nConstant, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image subtract constant.

- `NppStatus nppiSubC_32fc_C1IR` (const `Npp32fc` nConstant, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image subtract constant.

- `NppStatus nppiSubC_32fc_C3R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pConstants, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image subtract constant.

- `NppStatus nppiSubC_32fc_C3IR` (const `Npp32fc` *pConstants, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image subtract constant.

- `NppStatus nppiSubC_32fc_AC4R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pConstants, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha image subtract constant.

- `NppStatus nppiSubC_32fc_AC4IR` (const `Npp32fc` *pConstants, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha in place image subtract constant.

- `NppStatus nppiSubC_32fc_C4R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pConstants, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image subtract constant.

- `NppStatus nppiSubC_32fc_C4IR` (const `Npp32fc` *pConstants, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image subtract constant.

DivC

Divides each pixel of an image by a constant value.

- `NppStatus nppiDivC_8u_C1RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` nConstant, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_8u_C1IRSfs` (const `Npp8u` nConstant, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_8u_C3RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 8-bit unsigned char channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_8u_C3IRSfs` (const `Npp8u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 8-bit unsigned char channel 8-bit unsigned char in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_8u_AC4RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel with unmodified alpha image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_8u_AC4IRSfs` (const `Npp8u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel with unmodified alpha in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_8u_C4RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_8u_C4IRSfs` (const `Npp8u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16u_C1RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` nConstant, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16u_C1IRSfs` (const `Npp16u` nConstant, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16u_C3RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit unsigned short channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16u_C3IRSfs` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit unsigned short channel in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16u_AC4RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16u_AC4IRSfs` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16u_C4RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16u_C4IRSfs` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16s_C1RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` nConstant, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16s_C1IRSfs` (const `Npp16s` nConstant, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short channel in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16s_C3RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16s_C3IRSfs` (const `Npp16s` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short channel in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16s_AC4RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel with unmodified alpha image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16s_AC4IRSfs` (const `Npp16s` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel with unmodified alpha in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16s_C4RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16s_C4IRSfs` (const `Npp16s` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16sc_C1RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` nConstant, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16sc_C1IRSfs` (const `Npp16sc` nConstant, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16sc_C3RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` *pConstants, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16sc_C3IRSfs` (const `Npp16sc` *pConstants, `Npp16sc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image divided by constant, scale, then clamp to saturated value.

- `NppStatus nppiDivC_16sc_AC4RSfs` (const `Npp16sc` *pSrc1, int nSrc1Step, const `Npp16sc` *pConstants, `Npp16sc` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image divided by constant, scale, then clamp to saturated value.

- **NppStatus nppiDivC_16sc_AC4RSfs** (const **Npp16sc** *pConstants, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image divided by constant, scale, then clamp to saturated value.
- **NppStatus nppiDivC_32s_C1RSfs** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** nConstant, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer channel image divided by constant, scale, then clamp to saturated value.
- **NppStatus nppiDivC_32s_C1RSfs** (const **Npp32s** nConstant, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer channel in place image divided by constant, scale, then clamp to saturated value.
- **NppStatus nppiDivC_32s_C3RSfs** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pConstants, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed integer channel image divided by constant, scale, then clamp to saturated value.
- **NppStatus nppiDivC_32s_C3RSfs** (const **Npp32s** *pConstants, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed integer channel in place image divided by constant, scale, then clamp to saturated value.
- **NppStatus nppiDivC_32sc_C1RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** nConstant, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image divided by constant, scale, then clamp to saturated value.
- **NppStatus nppiDivC_32sc_C1RSfs** (const **Npp32sc** nConstant, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image divided by constant, scale, then clamp to saturated value.
- **NppStatus nppiDivC_32sc_C3RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pConstants, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image divided by constant, scale, then clamp to saturated value.
- **NppStatus nppiDivC_32sc_C3RSfs** (const **Npp32sc** *pConstants, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image divided by constant, scale, then clamp to saturated value.
- **NppStatus nppiDivC_32sc_AC4RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pConstants, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha image divided by constant, scale, then clamp to saturated value.
- **NppStatus nppiDivC_32sc_AC4RSfs** (const **Npp32sc** *pConstants, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image divided by constant, scale, then clamp to saturated value.

- **NppStatus nppiDivC_32f_C1R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** nConstant, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel image divided by constant.
- **NppStatus nppiDivC_32f_C1IR** (const **Npp32f** nConstant, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel in place image divided by constant.
- **NppStatus nppiDivC_32f_C3R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pConstants, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel image divided by constant.
- **NppStatus nppiDivC_32f_C3IR** (const **Npp32f** *pConstants, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel in place image divided by constant.
- **NppStatus nppiDivC_32f_AC4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pConstants, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel with unmodified alpha image divided by constant.
- **NppStatus nppiDivC_32f_AC4IR** (const **Npp32f** *pConstants, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel with unmodified alpha in place image divided by constant.
- **NppStatus nppiDivC_32f_C4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pConstants, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel image divided by constant.
- **NppStatus nppiDivC_32f_C4IR** (const **Npp32f** *pConstants, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel in place image divided by constant.
- **NppStatus nppiDivC_32fc_C1R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** nConstant, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image divided by constant.
- **NppStatus nppiDivC_32fc_C1IR** (const **Npp32fc** nConstant, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image divided by constant.
- **NppStatus nppiDivC_32fc_C3R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pConstants, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image divided by constant.
- **NppStatus nppiDivC_32fc_C3IR** (const **Npp32fc** *pConstants, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image divided by constant.

- **NppStatus** **nppiDivC_32fc_AC4R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pConstants, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha image divided by constant.
- **NppStatus** **nppiDivC_32fc_AC4IR** (const **Npp32fc** *pConstants, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha in place image divided by constant.
- **NppStatus** **nppiDivC_32fc_C4R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pConstants, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image divided by constant.
- **NppStatus** **nppiDivC_32fc_C4IR** (const **Npp32fc** *pConstants, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image divided by constant.

AbsDiffC

Determines absolute difference between each pixel of an image and a constant value.

- **NppStatus** **nppiAbsDiffC_8u_C1R** (const **Npp8u** *pSrc1, int nSrc1Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **Npp8u** nConstant)
One 8-bit unsigned char channel image absolute difference with constant.
- **NppStatus** **nppiAbsDiffC_16u_C1R** (const **Npp16u** *pSrc1, int nSrc1Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, **Npp32u** nConstant)
One 16-bit unsigned short channel image absolute difference with constant.
- **NppStatus** **nppiAbsDiffC_32f_C1R** (const **Npp32f** *pSrc1, int nSrc1Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI, **Npp32f** nConstant)
One 32-bit floating point channel image absolute difference with constant.

Add Image

Pixel by pixel addition of two images.

- **NppStatus** **nppiAdd_8u_C1RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus** **nppiAdd_8u_C1IRSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiAdd_8u_C3RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 8-bit unsigned char channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiAdd_8u_C3IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiAdd_8u_AC4RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel with unmodified alpha image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiAdd_8u_AC4IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel with unmodified alpha in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiAdd_8u_C4RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiAdd_8u_C4IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiAdd_16u_C1RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiAdd_16u_C1IRSfs` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiAdd_16u_C3RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit unsigned short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiAdd_16u_C3IRSfs` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_16u_AC4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel with unmodified alpha image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_16u_AC4IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel with unmodified alpha in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_16u_C4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_16u_C4IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_16s_C1RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_16s_C1IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_16s_C3RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_16s_C3IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_16s_AC4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_16s_AC4IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_16s_C4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit signed short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_16s_C4IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit signed short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_16sc_C1RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_16sc_C1IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_16sc_C3RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_16sc_C3IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_16sc_AC4RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_16sc_AC4IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_32s_C1RSfs** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 32-bit signed integer channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_32s_C1R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)

Note: This function is to be deprecated in future NPP releases, use the function above with a scale factor of 0 instead.

- **NppStatus nppiAdd_32s_C1IRSfs** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 32-bit signed integer channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiAdd_32s_C3RSfs** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed integer channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_32s_C3IRSfs** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_32sc_C1RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_32sc_C1IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_32sc_C3RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_32sc_C3IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_32sc_AC4RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_32sc_AC4IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiAdd_32f_C1R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel image addition.
- **NppStatus nppiAdd_32f_C1IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel in place image addition.
- **NppStatus nppiAdd_32f_C3R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel image addition.

- **NppStatus** **nppiAdd_32f_C3IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel in place image addition.
- **NppStatus** **nppiAdd_32f_AC4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel with unmodified alpha image addition.
- **NppStatus** **nppiAdd_32f_AC4IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel with unmodified alpha in place image addition.
- **NppStatus** **nppiAdd_32f_C4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel image addition.
- **NppStatus** **nppiAdd_32f_C4IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel in place image addition.
- **NppStatus** **nppiAdd_32fc_C1R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pSrc2, int nSrc2Step, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image addition.
- **NppStatus** **nppiAdd_32fc_C1IR** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image addition.
- **NppStatus** **nppiAdd_32fc_C3R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pSrc2, int nSrc2Step, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image addition.
- **NppStatus** **nppiAdd_32fc_C3IR** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image addition.
- **NppStatus** **nppiAdd_32fc_AC4R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pSrc2, int nSrc2Step, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image addition.
- **NppStatus** **nppiAdd_32fc_AC4IR** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image addition.
- **NppStatus** **nppiAdd_32fc_C4R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pSrc2, int nSrc2Step, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image addition.

- `NppStatus nppiAdd_32fc_C4IR` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image addition.

Add Square Image

Pixel by pixel addition of squared pixels from source image to floating point pixel values of destination image.

- `NppStatus nppiAddSquare_8u32f_C1IMR` (const `Npp8u` *pSrc, int nSrcStep, const `Npp8u` *pMask, int nMaskStep, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel image squared then added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

- `NppStatus nppiAddSquare_8u32f_C1IR` (const `Npp8u` *pSrc, int nSrcStep, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel image squared then added to in place floating point destination image.

- `NppStatus nppiAddSquare_16u32f_C1IMR` (const `Npp16u` *pSrc, int nSrcStep, const `Npp8u` *pMask, int nMaskStep, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel image squared then added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

- `NppStatus nppiAddSquare_16u32f_C1IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel image squared then added to in place floating point destination image.

- `NppStatus nppiAddSquare_32f_C1IMR` (const `Npp32f` *pSrc, int nSrcStep, const `Npp8u` *pMask, int nMaskStep, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 32-bit floating point channel image squared then added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

- `NppStatus nppiAddSquare_32f_C1IR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 32-bit floating point channel image squared then added to in place floating point destination image.

Add Product Image

Pixel by pixel addition of product of pixels from two source images to floating point pixel values of destination image.

- `NppStatus nppiAddProduct_8u32f_C1IMR` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, const `Npp8u` *pMask, int nMaskStep, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel image product added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

- `NppStatus nppiAddProduct_8u32f_C1IR` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel image product added to in place floating point destination image.

- **NppStatus** **nppiAddProduct_16u32f_C1IMR** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, const **Npp8u** *pMask, int nMaskStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 16-bit unsigned short channel image product added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

- **NppStatus** **nppiAddProduct_16u32f_C1IR** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 16-bit unsigned short channel image product added to in place floating point destination image.

- **NppStatus** **nppiAddProduct_32f_C1IMR** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, const **Npp8u** *pMask, int nMaskStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 32-bit floating point channel image product added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

- **NppStatus** **nppiAddProduct_32f_C1IR** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 32-bit floating point channel image product added to in place floating point destination image.

Add Weighted Image

Pixel by pixel addition of alpha weighted pixel values from a source image to floating point pixel values of destination image.

- **NppStatus** **nppiAddWeighted_8u32f_C1IMR** (const **Npp8u** *pSrc, int nSrcStep, const **Npp8u** *pMask, int nMaskStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **Npp32f** nAlpha)

One 8-bit unsigned char channel alpha weighted image added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

- **NppStatus** **nppiAddWeighted_8u32f_C1IR** (const **Npp8u** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **Npp32f** nAlpha)

One 8-bit unsigned char channel alpha weighted image added to in place floating point destination image.

- **NppStatus** **nppiAddWeighted_16u32f_C1IMR** (const **Npp16u** *pSrc, int nSrcStep, const **Npp8u** *pMask, int nMaskStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **Npp32f** nAlpha)

One 16-bit unsigned short channel alpha weighted image added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

- **NppStatus** **nppiAddWeighted_16u32f_C1IR** (const **Npp16u** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **Npp32f** nAlpha)

One 16-bit unsigned short channel alpha weighted image added to in place floating point destination image.

- **NppStatus** **nppiAddWeighted_32f_C1IMR** (const **Npp32f** *pSrc, int nSrcStep, const **Npp8u** *pMask, int nMaskStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **Npp32f** nAlpha)

One 32-bit floating point channel alpha weighted image added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

- `NppStatus nppiAddWeighted_32f_C1IR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, `Npp32f` nAlpha)

One 32-bit floating point channel alpha weighted image added to in place floating point destination image.

Mul Image

Pixel by pixel multiply of two images.

- `NppStatus nppiMul_8u_C1RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_8u_C1IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_8u_C3RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 8-bit unsigned char channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_8u_C3IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_8u_AC4RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel with unmodified alpha image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_8u_AC4IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel with unmodified alpha in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_8u_C4RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_8u_C4IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16u_C1RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16u_C1RSfs` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16u_C3RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit unsigned short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16u_C3IRSfs` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16u_AC4RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16u_AC4IRSfs` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16u_C4RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16u_C4IRSfs` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16s_C1RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pSrc2, int nSrc2Step, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16s_C1IRSfs` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 16-bit signed short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiMul_16s_C3RSfs` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp16s` *pSrc2, int nSrc2Step, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Three 16-bit signed short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiMul_16s_C3IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_16s_AC4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_16s_AC4IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_16s_C4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_16s_C4IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_16sc_C1RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_16sc_C1IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_16sc_C3RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_16sc_C3IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_16sc_AC4RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiMul_16sc_AC4IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_32s_C1RSfs** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_32s_C1R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Note: This function is to be deprecated in future NPP releases, use the function above with a scale factor of 0 instead.
- **NppStatus nppiMul_32s_C1IRSfs** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_32s_C3RSfs** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed integer channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_32s_C3IRSfs** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_32sc_C1RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_32sc_C1IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_32sc_C3RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_32sc_C3IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiMul_32sc_AC4RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiMul_32sc_AC4IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiMul_32f_C1R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

One 32-bit floating point channel image multiplication.

- **NppStatus** **nppiMul_32f_C1IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 32-bit floating point channel in place image multiplication.

- **NppStatus** **nppiMul_32f_C3R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

Three 32-bit floating point channel image multiplication.

- **NppStatus** **nppiMul_32f_C3IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 32-bit floating point channel in place image multiplication.

- **NppStatus** **nppiMul_32f_AC4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point channel with unmodified alpha image multiplication.

- **NppStatus** **nppiMul_32f_AC4IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point channel with unmodified alpha in place image multiplication.

- **NppStatus** **nppiMul_32f_C4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point channel image multiplication.

- **NppStatus** **nppiMul_32f_C4IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point channel in place image multiplication.

- **NppStatus** **nppiMul_32fc_C1R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pSrc2, int nSrc2Step, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image multiplication.

- **NppStatus** **nppiMul_32fc_C1IR** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image multiplication.

- **NppStatus** **nppiMul_32fc_C3R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pSrc2, int nSrc2Step, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)

Three 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image multiplication.

- **NppStatus nppiMul_32fc_C3IR** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image multiplication.

- **NppStatus nppiMul_32fc_AC4R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pSrc2, int nSrc2Step, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image multiplication.

- **NppStatus nppiMul_32fc_AC4IR** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image multiplication.

- **NppStatus nppiMul_32fc_C4R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pSrc2, int nSrc2Step, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image multiplication.

- **NppStatus nppiMul_32fc_C4IR** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image multiplication.

MulScale Image

Pixel by pixel multiplies each pixel of two images then scales the result by the maximum value for the data bit width.

- **NppStatus nppiMulScale_8u_C1R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)

One 8-bit unsigned char channel image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_8u_C1IR** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 8-bit unsigned char channel in place image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_8u_C3R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Three 8-bit unsigned char channel image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_8u_C3IR** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 8-bit unsigned char channel in place image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_8u_AC4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 8-bit unsigned char channel with unmodified alpha image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_8u_AC4IR** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 8-bit unsigned char channel with unmodified alpha in place image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_8u_C4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 8-bit unsigned char channel image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_8u_C4IR** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 8-bit unsigned char channel in place image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_16u_C1R** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)

One 16-bit unsigned short channel image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_16u_C1IR** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 16-bit unsigned short channel in place image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_16u_C3R** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Three 16-bit unsigned short channel image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_16u_C3IR** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 16-bit unsigned short channel in place image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_16u_AC4R** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 16-bit unsigned short channel with unmodified alpha image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_16u_AC4IR** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 16-bit unsigned short channel with unmodified alpha in place image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_16u_C4R** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 16-bit unsigned short channel image multiplication then scale by maximum value for pixel bit width.

- **NppStatus nppiMulScale_16u_C4IR** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 16-bit unsigned short channel in place image multiplication then scale by maximum value for pixel bit width.

Sub Image

Pixel by pixel subtraction of two images.

- `NppStatus nppiSub_8u_C1RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiSub_8u_C1IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiSub_8u_C3RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiSub_8u_C3IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiSub_8u_AC4RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel with unmodified alpha image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiSub_8u_AC4IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel with unmodified alpha in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiSub_8u_C4RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiSub_8u_C4IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiSub_16u_C1RSfs` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSub_16u_C1RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16u_C3RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16u_C3IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16u_AC4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel with unmodified alpha image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16u_AC4IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel with unmodified alpha in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16u_C4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16u_C4IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16s_C1RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16s_C1IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16s_C3RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSub_16s_C3IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16s_AC4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16s_AC4IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16s_C4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16s_C4IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16sc_C1RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16sc_C1IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16sc_C3RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16sc_C3IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16sc_AC4RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSub_16sc_AC4IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiSub_32s_C1RSfs** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 32-bit signed integer channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiSub_32s_C1R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)

Note: This function is to be deprecated in future NPP releases, use the function above with a scale factor of 0 instead.

- **NppStatus** **nppiSub_32s_C1IRSfs** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 32-bit signed integer channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiSub_32s_C3RSfs** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 32-bit signed integer channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiSub_32s_C3IRSfs** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 32-bit signed integer channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiSub_32sc_C1RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiSub_32sc_C1IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiSub_32sc_C3RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiSub_32sc_C3IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiSub_32sc_AC4RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiSub_32sc_AC4IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image subtraction, scale by $2^{\wedge}(-nScaleFactor)$, then clamp to saturated value.
- **NppStatus** **nppiSub_32f_C1R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel image subtraction.
- **NppStatus** **nppiSub_32f_C1IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel in place image subtraction.
- **NppStatus** **nppiSub_32f_C3R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel image subtraction.
- **NppStatus** **nppiSub_32f_C3IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel in place image subtraction.
- **NppStatus** **nppiSub_32f_AC4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel with unmodified alpha image subtraction.
- **NppStatus** **nppiSub_32f_AC4IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel with unmodified alpha in place image subtraction.
- **NppStatus** **nppiSub_32f_C4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel image subtraction.
- **NppStatus** **nppiSub_32f_C4IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel in place image subtraction.
- **NppStatus** **nppiSub_32fc_C1R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pSrc2, int nSrc2Step, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image subtraction.
- **NppStatus** **nppiSub_32fc_C1IR** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32fc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image subtraction.
- **NppStatus** **nppiSub_32fc_C3R** (const **Npp32fc** *pSrc1, int nSrc1Step, const **Npp32fc** *pSrc2, int nSrc2Step, **Npp32fc** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image subtraction.

- `NppStatus nppiSub_32fc_C3IR` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image subtraction.
- `NppStatus nppiSub_32fc_AC4R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pSrc2, int nSrc2Step, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image subtraction.
- `NppStatus nppiSub_32fc_AC4IR` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image subtraction.
- `NppStatus nppiSub_32fc_C4R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pSrc2, int nSrc2Step, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image subtraction.
- `NppStatus nppiSub_32fc_C4IR` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image subtraction.

Div Image

Pixel by pixel division of two images.

- `NppStatus nppiDiv_8u_C1RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiDiv_8u_C1IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiDiv_8u_C3RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiDiv_8u_C3IRSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- `NppStatus nppiDiv_8u_AC4RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel with unmodified alpha image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_8u_AC4RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel with unmodified alpha in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_8u_C4RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_8u_C4IRSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_16u_C1RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_16u_C1IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_16u_C3RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_16u_C3IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_16u_AC4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_16u_AC4IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_16u_C4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_16u_C4IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16s_C1RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16s_C1IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16s_C3RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16s_C3IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16s_AC4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16s_AC4IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel with unmodified alpha in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16s_C4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16s_C4IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16sc_C1RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16sc_C1IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_16sc_C3RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16sc_C3IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16sc_AC4RSfs** (const **Npp16sc** *pSrc1, int nSrc1Step, const **Npp16sc** *pSrc2, int nSrc2Step, **Npp16sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_16sc_AC4IRSfs** (const **Npp16sc** *pSrc, int nSrcStep, **Npp16sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_32s_C1RSfs** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_32s_C1R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Note: This function is to be deprecated in future NPP releases, use the function above with a scale factor of 0 instead.
- **NppStatus nppiDiv_32s_C1IRSfs** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_32s_C3RSfs** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed integer channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_32s_C3IRSfs** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_32sc_C1RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_32sc_C1IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_32sc_C3RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_32sc_C3IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_32sc_AC4RSfs** (const **Npp32sc** *pSrc1, int nSrc1Step, const **Npp32sc** *pSrc2, int nSrc2Step, **Npp32sc** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_32sc_AC4IRSfs** (const **Npp32sc** *pSrc, int nSrcStep, **Npp32sc** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_32f_C1R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel image division.
- **NppStatus nppiDiv_32f_C1IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel in place image division.
- **NppStatus nppiDiv_32f_C3R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel image division.
- **NppStatus nppiDiv_32f_C3IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel in place image division.
- **NppStatus nppiDiv_32f_AC4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel with unmodified alpha image division.
- **NppStatus nppiDiv_32f_AC4IR** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel with unmodified alpha in place image division.
- **NppStatus nppiDiv_32f_C4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point channel image division.

- `NppStatus nppiDiv_32f_C4IR` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel in place image division.

- `NppStatus nppiDiv_32fc_C1R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pSrc2, int nSrc2Step, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image division.

- `NppStatus nppiDiv_32fc_C1IR` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image division.

- `NppStatus nppiDiv_32fc_C3R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pSrc2, int nSrc2Step, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image division.

- `NppStatus nppiDiv_32fc_C3IR` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image division.

- `NppStatus nppiDiv_32fc_AC4R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pSrc2, int nSrc2Step, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image division.

- `NppStatus nppiDiv_32fc_AC4IR` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image division.

- `NppStatus nppiDiv_32fc_C4R` (const `Npp32fc` *pSrc1, int nSrc1Step, const `Npp32fc` *pSrc2, int nSrc2Step, `Npp32fc` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image division.

- `NppStatus nppiDiv_32fc_C4IR` (const `Npp32fc` *pSrc, int nSrcStep, `Npp32fc` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image division.

Div_Round Image

Pixel by pixel division of two images using result rounding modes.

- `NppStatus nppiDiv_Round_8u_C1RSfs` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppRoundMode` rndMode, int nScaleFactor)

One 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_Round_8u_C1IRSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
One 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_8u_C3RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Three 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_8u_C3IRSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Three 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_8u_AC4RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Four 8-bit unsigned char channel image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_8u_AC4IRSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Four 8-bit unsigned char channel in place image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_8u_C4RSfs** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Four 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_8u_C4IRSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Four 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16u_C1RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
One 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16u_C1IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
One 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16u_C3RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Three 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiDiv_Round_16u_C3RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Three 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16u_AC4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Four 16-bit unsigned short channel image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16u_AC4IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Four 16-bit unsigned short channel in place image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16u_C4RSfs** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Four 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16u_C4IRSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Four 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16s_C1RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
One 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16s_C1IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
One 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16s_C3RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Three 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16s_C3IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)
Three 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiDiv_Round_16s_AC4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)

Four 16-bit signed short channel image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiDiv_Round_16s_AC4IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)

Four 16-bit signed short channel in place image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiDiv_Round_16s_C4RSfs** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)

Four 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiDiv_Round_16s_C4IRSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, **NppRoundMode** rndMode, int nScaleFactor)

Four 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Abs Image

Absolute value of each pixel value in an image.

- **NppStatus** **nppiAbs_16s_C1R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)

One 16-bit signed short channel image absolute value.

- **NppStatus** **nppiAbs_16s_C1IR** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 16-bit signed short channel in place image absolute value.

- **NppStatus** **nppiAbs_16s_C3R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)

Three 16-bit signed short channel image absolute value.

- **NppStatus** **nppiAbs_16s_C3IR** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Three 16-bit signed short channel in place image absolute value.

- **NppStatus** **nppiAbs_16s_AC4R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 16-bit signed short channel image absolute value with unmodified alpha.

- **NppStatus** **nppiAbs_16s_AC4IR** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 16-bit signed short channel in place image absolute value with unmodified alpha.

- **NppStatus** **nppiAbs_16s_C4R** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 16-bit signed short channel image absolute value.

- **NppStatus** **nppiAbs_16s_C4IR** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 16-bit signed short channel in place image absolute value.

- **NppStatus nppiAbs_32f_C1R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel image absolute value.
- **NppStatus nppiAbs_32f_C1IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel in place image absolute value.
- **NppStatus nppiAbs_32f_C3R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel image absolute value.
- **NppStatus nppiAbs_32f_C3IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel in place image absolute value.
- **NppStatus nppiAbs_32f_AC4R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel image absolute value with unmodified alpha.
- **NppStatus nppiAbs_32f_AC4IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel in place image absolute value with unmodified alpha.
- **NppStatus nppiAbs_32f_C4R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel image absolute value.
- **NppStatus nppiAbs_32f_C4IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit floating point channel in place image absolute value.

AbsDiff Image

Pixel by pixel absolute difference between two images.

- **NppStatus nppiAbsDiff_8u_C1R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel absolute difference of image1 minus image2.
- **NppStatus nppiAbsDiff_16u_C1R** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 16-bit unsigned short channel absolute difference of image1 minus image2.
- **NppStatus nppiAbsDiff_32f_C1R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel absolute difference of image1 minus image2.

Sqr Image

Square each pixel in an image.

- **NppStatus nppiSqr_8u_C1RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_8u_C1IRSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_8u_C3RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_8u_C3IRSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_8u_AC4RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_8u_AC4IRSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel in place image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_8u_C4RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_8u_C4IRSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16u_C1RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16u_C1IRSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16u_C3RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSqr_16u_C3IRSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16u_AC4RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16u_AC4IRSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel in place image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16u_C4RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16u_C4IRSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16s_C1RSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16s_C1IRSfs** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16s_C3RSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16s_C3IRSfs** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16s_AC4RSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit signed short channel image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqr_16s_AC4IRSfs** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit signed short channel in place image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiSqr_16s_C4RSfs` (const `Npp16s` *pSrc, int nSrcStep, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiSqr_16s_C4IRSfs` (`Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI, int nScaleFactor)

Four 16-bit signed short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- `NppStatus nppiSqr_32f_C1R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 32-bit floating point channel image squared.

- `NppStatus nppiSqr_32f_C1IR` (`Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 32-bit floating point channel in place image squared.

- `NppStatus nppiSqr_32f_C3R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 32-bit floating point channel image squared.

- `NppStatus nppiSqr_32f_C3IR` (`Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 32-bit floating point channel in place image squared.

- `NppStatus nppiSqr_32f_AC4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel image squared with unmodified alpha.

- `NppStatus nppiSqr_32f_AC4IR` (`Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel in place image squared with unmodified alpha.

- `NppStatus nppiSqr_32f_C4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel image squared.

- `NppStatus nppiSqr_32f_C4IR` (`Npp32f` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit floating point channel in place image squared.

Sqrt Image

Pixel by pixel square root of each pixel in an image.

- `NppStatus nppiSqrt_8u_C1RSfs` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSqrt_8u_C1RSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 8-bit unsigned char channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqrt_8u_C3RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqrt_8u_C3IRSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqrt_8u_AC4RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqrt_8u_AC4IRSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 8-bit unsigned char channel in place image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqrt_16u_C1RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqrt_16u_C1IRSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqrt_16u_C3RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqrt_16u_C3IRSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqrt_16u_AC4RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Four 16-bit unsigned short channel image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiSqrt_16u_AC4IRSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel in place image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSqrt_16s_C1RSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit signed short channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSqrt_16s_C1IRSfs** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit signed short channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSqrt_16s_C3RSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 16-bit signed short channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSqrt_16s_C3IRSfs** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 16-bit signed short channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSqrt_16s_AC4RSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit signed short channel image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSqrt_16s_AC4IRSfs** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Four 16-bit signed short channel in place image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiSqrt_32f_C1R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

One 32-bit floating point channel image square root.

- **NppStatus nppiSqrt_32f_C1IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 32-bit floating point channel in place image square root.

- **NppStatus nppiSqrt_32f_C3R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

Three 32-bit floating point channel image square root.

- **NppStatus nppiSqrt_32f_C3IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Three 32-bit floating point channel in place image square root.

- **NppStatus nppiSqrt_32f_AC4R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point channel image square root with unmodified alpha.

- **NppStatus nppiSqrt_32f_AC4IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point channel in place image square root with unmodified alpha.

- **NppStatus nppiSqrt_32f_C4R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point channel image square root.

- **NppStatus nppiSqrt_32f_C4IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 32-bit floating point channel in place image square root.

Ln Image

Pixel by pixel natural logarithm of each pixel in an image.

- **NppStatus nppiLn_8u_C1RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiLn_8u_C1RSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiLn_8u_C3RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 8-bit unsigned char channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiLn_8u_C3RSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 8-bit unsigned char channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiLn_16u_C1RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiLn_16u_C1RSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiLn_16u_C3RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 16-bit unsigned short channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiLn_16u_C3RSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 16-bit unsigned short channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiLn_16s_C1RSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit signed short channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiLn_16s_C1IRSfs** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 16-bit signed short channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiLn_16s_C3RSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 16-bit signed short channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiLn_16s_C3IRSfs** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

Three 16-bit signed short channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiLn_32f_C1R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

One 32-bit floating point channel image natural logarithm.

- **NppStatus** **nppiLn_32f_C1IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

One 32-bit floating point channel in place image natural logarithm.

- **NppStatus** **nppiLn_32f_C3R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

Three 32-bit floating point channel image natural logarithm.

- **NppStatus** **nppiLn_32f_C3IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Three 32-bit floating point channel in place image natural logarithm.

Exp Image

Exponential value of each pixel in an image.

- **NppStatus** **nppiExp_8u_C1RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus** **nppiExp_8u_C1IRSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiExp_8u_C3RSfs** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiExp_8u_C3IRSfs** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 8-bit unsigned char channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiExp_16u_C1RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiExp_16u_C1IRSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit unsigned short channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiExp_16u_C3RSfs** (const **Npp16u** *pSrc, int nSrcStep, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiExp_16u_C3IRSfs** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit unsigned short channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiExp_16s_C1RSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiExp_16s_C1IRSfs** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
One 16-bit signed short channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiExp_16s_C3RSfs** (const **Npp16s** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.
- **NppStatus nppiExp_16s_C3IRSfs** (**Npp16s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI, int nScaleFactor)
Three 16-bit signed short channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

- **NppStatus nppiExp_32f_C1R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel image exponential.
- **NppStatus nppiExp_32f_C1IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit floating point channel in place image exponential.
- **NppStatus nppiExp_32f_C3R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel image exponential.
- **NppStatus nppiExp_32f_C3IR** (**Npp32f** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 32-bit floating point channel in place image exponential.

AndC Image

Pixel by pixel logical and of an image with a constant.

- **NppStatus nppiAndC_8u_C1R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** nConstant, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel image logical and with constant.
- **NppStatus nppiAndC_8u_C1IR** (const **Npp8u** nConstant, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel in place image logical and with constant.
- **NppStatus nppiAndC_8u_C3R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel image logical and with constant.
- **NppStatus nppiAndC_8u_C3IR** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel in place image logical and with constant.
- **NppStatus nppiAndC_8u_AC4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 8-bit unsigned char channel image logical and with constant with unmodified alpha.
- **NppStatus nppiAndC_8u_AC4IR** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 8-bit unsigned char channel in place image logical and with constant with unmodified alpha.
- **NppStatus nppiAndC_8u_C4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 8-bit unsigned char channel image logical and with constant.
- **NppStatus nppiAndC_8u_C4IR** (const **Npp8u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 8-bit unsigned char channel in place image logical and with constant.

- `NppStatus nppiAndC_16u_C1R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` nConstant, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
One 16-bit unsigned short channel image logical and with constant.
- `NppStatus nppiAndC_16u_C1IR` (const `Npp16u` nConstant, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 16-bit unsigned short channel in place image logical and with constant.
- `NppStatus nppiAndC_16u_C3R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 16-bit unsigned short channel image logical and with constant.
- `NppStatus nppiAndC_16u_C3IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 16-bit unsigned short channel in place image logical and with constant.
- `NppStatus nppiAndC_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel image logical and with constant with unmodified alpha.
- `NppStatus nppiAndC_16u_AC4IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel in place image logical and with constant with unmodified alpha.
- `NppStatus nppiAndC_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel image logical and with constant.
- `NppStatus nppiAndC_16u_C4IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel in place image logical and with constant.
- `NppStatus nppiAndC_32s_C1R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` nConstant, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
One 32-bit signed integer channel image logical and with constant.
- `NppStatus nppiAndC_32s_C1IR` (const `Npp32s` nConstant, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 32-bit signed integer channel in place image logical and with constant.
- `NppStatus nppiAndC_32s_C3R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 32-bit signed integer channel image logical and with constant.
- `NppStatus nppiAndC_32s_C3IR` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 32-bit signed integer channel in place image logical and with constant.
- `NppStatus nppiAndC_32s_AC4R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel image logical and with constant with unmodified alpha.

- `NppStatus nppiAndC_32s_AC4IR` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel in place image logical and with constant with unmodified alpha.

- `NppStatus nppiAndC_32s_C4R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel image logical and with constant.

- `NppStatus nppiAndC_32s_C4IR` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel in place image logical and with constant.

OrC Image

Pixel by pixel logical or of an image with a constant.

- `NppStatus nppiOrC_8u_C1R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` nConstant, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel image logical or with constant.

- `NppStatus nppiOrC_8u_C1IR` (const `Npp8u` nConstant, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel in place image logical or with constant.

- `NppStatus nppiOrC_8u_C3R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 8-bit unsigned char channel image logical or with constant.

- `NppStatus nppiOrC_8u_C3IR` (const `Npp8u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 8-bit unsigned char channel in place image logical or with constant.

- `NppStatus nppiOrC_8u_AC4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image logical or with constant with unmodified alpha.

- `NppStatus nppiOrC_8u_AC4IR` (const `Npp8u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image logical or with constant with unmodified alpha.

- `NppStatus nppiOrC_8u_C4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image logical or with constant.

- `NppStatus nppiOrC_8u_C4IR` (const `Npp8u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image logical or with constant.

- `NppStatus nppiOrC_16u_C1R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` nConstant, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
One 16-bit unsigned short channel image logical or with constant.
- `NppStatus nppiOrC_16u_C1IR` (const `Npp16u` nConstant, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 16-bit unsigned short channel in place image logical or with constant.
- `NppStatus nppiOrC_16u_C3R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 16-bit unsigned short channel image logical or with constant.
- `NppStatus nppiOrC_16u_C3IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 16-bit unsigned short channel in place image logical or with constant.
- `NppStatus nppiOrC_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel image logical or with constant with unmodified alpha.
- `NppStatus nppiOrC_16u_AC4IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel in place image logical or with constant with unmodified alpha.
- `NppStatus nppiOrC_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel image logical or with constant.
- `NppStatus nppiOrC_16u_C4IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel in place image logical or with constant.
- `NppStatus nppiOrC_32s_C1R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` nConstant, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
One 32-bit signed integer channel image logical or with constant.
- `NppStatus nppiOrC_32s_C1IR` (const `Npp32s` nConstant, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 32-bit signed integer channel in place image logical or with constant.
- `NppStatus nppiOrC_32s_C3R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 32-bit signed integer channel image logical or with constant.
- `NppStatus nppiOrC_32s_C3IR` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 32-bit signed integer channel in place image logical or with constant.
- `NppStatus nppiOrC_32s_AC4R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel image logical or with constant with unmodified alpha.

- `NppStatus nppiOrC_32s_AC4IR` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel in place image logical or with constant with unmodified alpha.

- `NppStatus nppiOrC_32s_C4R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel image logical or with constant.

- `NppStatus nppiOrC_32s_C4IR` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel in place image logical or with constant.

XorC Image

Pixel by pixel logical exclusive or of an image with a constant.

- `NppStatus nppiXorC_8u_C1R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` nConstant, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel image logical exclusive or with constant.

- `NppStatus nppiXorC_8u_C1IR` (const `Npp8u` nConstant, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel in place image logical exclusive or with constant.

- `NppStatus nppiXorC_8u_C3R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 8-bit unsigned char channel image logical exclusive or with constant.

- `NppStatus nppiXorC_8u_C3IR` (const `Npp8u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 8-bit unsigned char channel in place image logical exclusive or with constant.

- `NppStatus nppiXorC_8u_AC4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image logical exclusive or with constant with unmodified alpha.

- `NppStatus nppiXorC_8u_AC4IR` (const `Npp8u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image logical exclusive or with constant with unmodified alpha.

- `NppStatus nppiXorC_8u_C4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image logical exclusive or with constant.

- `NppStatus nppiXorC_8u_C4IR` (const `Npp8u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image logical exclusive or with constant.

- `NppStatus nppiXorC_16u_C1R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` nConstant, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
One 16-bit unsigned short channel image logical exclusive or with constant.
- `NppStatus nppiXorC_16u_C1IR` (const `Npp16u` nConstant, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 16-bit unsigned short channel in place image logical exclusive or with constant.
- `NppStatus nppiXorC_16u_C3R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 16-bit unsigned short channel image logical exclusive or with constant.
- `NppStatus nppiXorC_16u_C3IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 16-bit unsigned short channel in place image logical exclusive or with constant.
- `NppStatus nppiXorC_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel image logical exclusive or with constant with unmodified alpha.
- `NppStatus nppiXorC_16u_AC4IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel in place image logical exclusive or with constant with unmodified alpha.
- `NppStatus nppiXorC_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel image logical exclusive or with constant.
- `NppStatus nppiXorC_16u_C4IR` (const `Npp16u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 16-bit unsigned short channel in place image logical exclusive or with constant.
- `NppStatus nppiXorC_32s_C1R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` nConstant, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
One 32-bit signed integer channel image logical exclusive or with constant.
- `NppStatus nppiXorC_32s_C1IR` (const `Npp32s` nConstant, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 32-bit signed integer channel in place image logical exclusive or with constant.
- `NppStatus nppiXorC_32s_C3R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 32-bit signed integer channel image logical exclusive or with constant.
- `NppStatus nppiXorC_32s_C3IR` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 32-bit signed integer channel in place image logical exclusive or with constant.
- `NppStatus nppiXorC_32s_AC4R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel image logical exclusive or with constant with unmodified alpha.

- `NppStatus nppiXorC_32s_AC4IR` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel in place image logical exclusive or with constant with unmodified alpha.

- `NppStatus nppiXorC_32s_C4R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pConstants, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel image logical exclusive or with constant.

- `NppStatus nppiXorC_32s_C4IR` (const `Npp32s` *pConstants, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 32-bit signed integer channel in place image logical exclusive or with constant.

RShiftC Image

Pixel by pixel right shift of an image by a constant value.

- `NppStatus nppiRShiftC_8u_C1R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp32u` nConstant, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel image right shift by constant.

- `NppStatus nppiRShiftC_8u_C1IR` (const `Npp32u` nConstant, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel in place image right shift by constant.

- `NppStatus nppiRShiftC_8u_C3R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 8-bit unsigned char channel image right shift by constant.

- `NppStatus nppiRShiftC_8u_C3IR` (const `Npp32u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 8-bit unsigned char channel in place image right shift by constant.

- `NppStatus nppiRShiftC_8u_AC4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image right shift by constant with unmodified alpha.

- `NppStatus nppiRShiftC_8u_AC4IR` (const `Npp32u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image right shift by constant with unmodified alpha.

- `NppStatus nppiRShiftC_8u_C4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image right shift by constant.

- `NppStatus nppiRShiftC_8u_C4IR` (const `Npp32u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image right shift by constant.

- `NppStatus nppiRShiftC_8s_C1R` (const `Npp8s` *pSrc1, int nSrc1Step, const `Npp32u` nConstant, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
One 8-bit signed char channel image right shift by constant.
- `NppStatus nppiRShiftC_8s_C1IR` (const `Npp32u` nConstant, `Npp8s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 8-bit signed char channel in place image right shift by constant.
- `NppStatus nppiRShiftC_8s_C3R` (const `Npp8s` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 8-bit signed char channel image right shift by constant.
- `NppStatus nppiRShiftC_8s_C3IR` (const `Npp32u` *pConstants, `Npp8s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 8-bit signed char channel in place image right shift by constant.
- `NppStatus nppiRShiftC_8s_AC4R` (const `Npp8s` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 8-bit signed char channel image right shift by constant with unmodified alpha.
- `NppStatus nppiRShiftC_8s_AC4IR` (const `Npp32u` *pConstants, `Npp8s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 8-bit signed char channel in place image right shift by constant with unmodified alpha.
- `NppStatus nppiRShiftC_8s_C4R` (const `Npp8s` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp8s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 8-bit signed char channel image right shift by constant.
- `NppStatus nppiRShiftC_8s_C4IR` (const `Npp32u` *pConstants, `Npp8s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 8-bit signed char channel in place image right shift by constant.
- `NppStatus nppiRShiftC_16u_C1R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp32u` nConstant, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
One 16-bit unsigned short channel image right shift by constant.
- `NppStatus nppiRShiftC_16u_C1IR` (const `Npp32u` nConstant, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 16-bit unsigned short channel in place image right shift by constant.
- `NppStatus nppiRShiftC_16u_C3R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 16-bit unsigned short channel image right shift by constant.
- `NppStatus nppiRShiftC_16u_C3IR` (const `Npp32u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 16-bit unsigned short channel in place image right shift by constant.
- `NppStatus nppiRShiftC_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image right shift by constant with unmodified alpha.

- `NppStatus nppiRShiftC_16u_AC4IR` (const `Npp32u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image right shift by constant with unmodified alpha.

- `NppStatus nppiRShiftC_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image right shift by constant.

- `NppStatus nppiRShiftC_16u_C4IR` (const `Npp32u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image right shift by constant.

- `NppStatus nppiRShiftC_16s_C1R` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp32u` nConstant, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 16-bit signed short channel image right shift by constant.

- `NppStatus nppiRShiftC_16s_C1IR` (const `Npp32u` nConstant, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 16-bit signed short channel in place image right shift by constant.

- `NppStatus nppiRShiftC_16s_C3R` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 16-bit signed short channel image right shift by constant.

- `NppStatus nppiRShiftC_16s_C3IR` (const `Npp32u` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 16-bit signed short channel in place image right shift by constant.

- `NppStatus nppiRShiftC_16s_AC4R` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit signed short channel image right shift by constant with unmodified alpha.

- `NppStatus nppiRShiftC_16s_AC4IR` (const `Npp32u` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit signed short channel in place image right shift by constant with unmodified alpha.

- `NppStatus nppiRShiftC_16s_C4R` (const `Npp16s` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit signed short channel image right shift by constant.

- `NppStatus nppiRShiftC_16s_C4IR` (const `Npp32u` *pConstants, `Npp16s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit signed short channel in place image right shift by constant.

- `NppStatus nppiRShiftC_32s_C1R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32u` nConstant, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 32-bit signed integer channel image right shift by constant.

- **NppStatus nppiRShiftC_32s_C1IR** (const **Npp32u** nConstant, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit signed integer channel in place image right shift by constant.
- **NppStatus nppiRShiftC_32s_C3R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32u** *pConstants, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit signed integer channel image right shift by constant.
- **NppStatus nppiRShiftC_32s_C3IR** (const **Npp32u** *pConstants, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 32-bit signed integer channel in place image right shift by constant.
- **NppStatus nppiRShiftC_32s_AC4R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32u** *pConstants, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel image right shift by constant with unmodified alpha.
- **NppStatus nppiRShiftC_32s_AC4IR** (const **Npp32u** *pConstants, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel in place image right shift by constant with unmodified alpha.
- **NppStatus nppiRShiftC_32s_C4R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32u** *pConstants, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel image right shift by constant.
- **NppStatus nppiRShiftC_32s_C4IR** (const **Npp32u** *pConstants, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel in place image right shift by constant.

LShiftC Image

Pixel by pixel left shift of an image by a constant value.

- **NppStatus nppiLShiftC_8u_C1R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp32u** nConstant, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel image left shift by constant.
- **NppStatus nppiLShiftC_8u_C1IR** (const **Npp32u** nConstant, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel in place image left shift by constant.
- **NppStatus nppiLShiftC_8u_C3R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp32u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel image left shift by constant.
- **NppStatus nppiLShiftC_8u_C3IR** (const **Npp32u** *pConstants, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel in place image left shift by constant.
- **NppStatus nppiLShiftC_8u_AC4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp32u** *pConstants, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 8-bit unsigned char channel image left shift by constant with unmodified alpha.

- `NppStatus nppiLShiftC_8u_AC4IR` (const `Npp32u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image left shift by constant with unmodified alpha.

- `NppStatus nppiLShiftC_8u_C4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image left shift by constant.

- `NppStatus nppiLShiftC_8u_C4IR` (const `Npp32u` *pConstants, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image left shift by constant.

- `NppStatus nppiLShiftC_16u_C1R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp32u` nConstant, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel image left shift by constant.

- `NppStatus nppiLShiftC_16u_C1IR` (const `Npp32u` nConstant, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel in place image left shift by constant.

- `NppStatus nppiLShiftC_16u_C3R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel image left shift by constant.

- `NppStatus nppiLShiftC_16u_C3IR` (const `Npp32u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel in place image left shift by constant.

- `NppStatus nppiLShiftC_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image left shift by constant with unmodified alpha.

- `NppStatus nppiLShiftC_16u_AC4IR` (const `Npp32u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image left shift by constant with unmodified alpha.

- `NppStatus nppiLShiftC_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp32u` *pConstants, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image left shift by constant.

- `NppStatus nppiLShiftC_16u_C4IR` (const `Npp32u` *pConstants, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image left shift by constant.

- `NppStatus nppiLShiftC_32s_C1R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32u` nConstant, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 32-bit signed integer channel image left shift by constant.

- **NppStatus nppiLShiftC_32s_C1IR** (const **Npp32u** nConstant, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit signed integer channel in place image left shift by constant.
- **NppStatus nppiLShiftC_32s_C3R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32u** *pConstants, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit signed integer channel image left shift by constant.
- **NppStatus nppiLShiftC_32s_C3IR** (const **Npp32u** *pConstants, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 32-bit signed integer channel in place image left shift by constant.
- **NppStatus nppiLShiftC_32s_AC4R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32u** *pConstants, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel image left shift by constant with unmodified alpha.
- **NppStatus nppiLShiftC_32s_AC4IR** (const **Npp32u** *pConstants, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel in place image left shift by constant with unmodified alpha.
- **NppStatus nppiLShiftC_32s_C4R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32u** *pConstants, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel image left shift by constant.
- **NppStatus nppiLShiftC_32s_C4IR** (const **Npp32u** *pConstants, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel in place image left shift by constant.

And Image

Pixel by pixel logical and of images.

- **NppStatus nppiAnd_8u_C1R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel image logical and.
- **NppStatus nppiAnd_8u_C1IR** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel in place image logical and.
- **NppStatus nppiAnd_8u_C3R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel image logical and.
- **NppStatus nppiAnd_8u_C3IR** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel in place image logical and.
- **NppStatus nppiAnd_8u_AC4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 8-bit unsigned char channel image logical and with unmodified alpha.

- `NppStatus nppiAnd_8u_AC4IR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image logical and with unmodified alpha.

- `NppStatus nppiAnd_8u_C4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image logical and.

- `NppStatus nppiAnd_8u_C4IR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image logical and.

- `NppStatus nppiAnd_16u_C1R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel image logical and.

- `NppStatus nppiAnd_16u_C1IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel in place image logical and.

- `NppStatus nppiAnd_16u_C3R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel image logical and.

- `NppStatus nppiAnd_16u_C3IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel in place image logical and.

- `NppStatus nppiAnd_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image logical and with unmodified alpha.

- `NppStatus nppiAnd_16u_AC4IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image logical and with unmodified alpha.

- `NppStatus nppiAnd_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image logical and.

- `NppStatus nppiAnd_16u_C4IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image logical and.

- `NppStatus nppiAnd_32s_C1R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pSrc2, int nSrc2Step, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 32-bit signed integer channel image logical and.

- **NppStatus nppiAnd_32s_C1IR** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit signed integer channel in place image logical and.
- **NppStatus nppiAnd_32s_C3R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit signed integer channel image logical and.
- **NppStatus nppiAnd_32s_C3IR** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 32-bit signed integer channel in place image logical and.
- **NppStatus nppiAnd_32s_AC4R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel image logical and with unmodified alpha.
- **NppStatus nppiAnd_32s_AC4IR** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel in place image logical and with unmodified alpha.
- **NppStatus nppiAnd_32s_C4R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel image logical and.
- **NppStatus nppiAnd_32s_C4IR** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel in place image logical and.

Or Image

Pixel by pixel logical or of images.

- **NppStatus nppiOr_8u_C1R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel image logical or.
- **NppStatus nppiOr_8u_C1IR** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel in place image logical or.
- **NppStatus nppiOr_8u_C3R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel image logical or.
- **NppStatus nppiOr_8u_C3IR** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel in place image logical or.
- **NppStatus nppiOr_8u_AC4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 8-bit unsigned char channel image logical or with unmodified alpha.

- `NppStatus nppiOr_8u_AC4IR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image logical or with unmodified alpha.

- `NppStatus nppiOr_8u_C4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image logical or.

- `NppStatus nppiOr_8u_C4IR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image logical or.

- `NppStatus nppiOr_16u_C1R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel image logical or.

- `NppStatus nppiOr_16u_C1IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel in place image logical or.

- `NppStatus nppiOr_16u_C3R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel image logical or.

- `NppStatus nppiOr_16u_C3IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel in place image logical or.

- `NppStatus nppiOr_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image logical or with unmodified alpha.

- `NppStatus nppiOr_16u_AC4IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image logical or with unmodified alpha.

- `NppStatus nppiOr_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image logical or.

- `NppStatus nppiOr_16u_C4IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image logical or.

- `NppStatus nppiOr_32s_C1R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pSrc2, int nSrc2Step, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 32-bit signed integer channel image logical or.

- `NppStatus nppiOr_32s_C1IR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 32-bit signed integer channel in place image logical or.
- `NppStatus nppiOr_32s_C3R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pSrc2, int nSrc2Step, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 32-bit signed integer channel image logical or.
- `NppStatus nppiOr_32s_C3IR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 32-bit signed integer channel in place image logical or.
- `NppStatus nppiOr_32s_AC4R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pSrc2, int nSrc2Step, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 32-bit signed integer channel image logical or with unmodified alpha.
- `NppStatus nppiOr_32s_AC4IR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 32-bit signed integer channel in place image logical or with unmodified alpha.
- `NppStatus nppiOr_32s_C4R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pSrc2, int nSrc2Step, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)
Four 32-bit signed integer channel image logical or.
- `NppStatus nppiOr_32s_C4IR` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Four 32-bit signed integer channel in place image logical or.

Xor Image

Pixel by pixel logical exclusive or of images.

- `NppStatus nppiXor_8u_C1R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
One 8-bit unsigned char channel image logical exclusive or.
- `NppStatus nppiXor_8u_C1IR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
One 8-bit unsigned char channel in place image logical exclusive or.
- `NppStatus nppiXor_8u_C3R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)
Three 8-bit unsigned char channel image logical exclusive or.
- `NppStatus nppiXor_8u_C3IR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)
Three 8-bit unsigned char channel in place image logical exclusive or.
- `NppStatus nppiXor_8u_AC4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image logical exclusive or with unmodified alpha.

- `NppStatus nppiXor_8u_AC4IR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image logical exclusive or with unmodified alpha.

- `NppStatus nppiXor_8u_C4R` (const `Npp8u` *pSrc1, int nSrc1Step, const `Npp8u` *pSrc2, int nSrc2Step, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image logical exclusive or.

- `NppStatus nppiXor_8u_C4IR` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image logical exclusive or.

- `NppStatus nppiXor_16u_C1R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel image logical exclusive or.

- `NppStatus nppiXor_16u_C1IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel in place image logical exclusive or.

- `NppStatus nppiXor_16u_C3R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel image logical exclusive or.

- `NppStatus nppiXor_16u_C3IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel in place image logical exclusive or.

- `NppStatus nppiXor_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image logical exclusive or with unmodified alpha.

- `NppStatus nppiXor_16u_AC4IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image logical exclusive or with unmodified alpha.

- `NppStatus nppiXor_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image logical exclusive or.

- `NppStatus nppiXor_16u_C4IR` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image logical exclusive or.

- `NppStatus nppiXor_32s_C1R` (const `Npp32s` *pSrc1, int nSrc1Step, const `Npp32s` *pSrc2, int nSrc2Step, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 32-bit signed integer channel image logical exclusive or.

- **NppStatus nppiXor_32s_C1IR** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 32-bit signed integer channel in place image logical exclusive or.
- **NppStatus nppiXor_32s_C3R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 32-bit signed integer channel image logical exclusive or.
- **NppStatus nppiXor_32s_C3IR** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 32-bit signed integer channel in place image logical exclusive or.
- **NppStatus nppiXor_32s_AC4R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel image logical exclusive or with unmodified alpha.
- **NppStatus nppiXor_32s_AC4IR** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel in place image logical exclusive or with unmodified alpha.
- **NppStatus nppiXor_32s_C4R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel image logical exclusive or.
- **NppStatus nppiXor_32s_C4IR** (const **Npp32s** *pSrc, int nSrcStep, **Npp32s** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Four 32-bit signed integer channel in place image logical exclusive or.

Not Image

Pixel by pixel logical not of image.

- **NppStatus nppiNot_8u_C1R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel image logical not.
- **NppStatus nppiNot_8u_C1IR** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
One 8-bit unsigned char channel in place image logical not.
- **NppStatus nppiNot_8u_C3R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel image logical not.
- **NppStatus nppiNot_8u_C3IR** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)
Three 8-bit unsigned char channel in place image logical not.
- **NppStatus nppiNot_8u_AC4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
Four 8-bit unsigned char channel image logical not with unmodified alpha.

- `NppStatus nppiNot_8u_AC4IR` (`Npp8u *pSrcDst`, `int nSrcDstStep`, `NppiSize oSizeROI`)
Four 8-bit unsigned char channel in place image logical not with unmodified alpha.
- `NppStatus nppiNot_8u_C4R` (`const Npp8u *pSrc`, `int nSrcStep`, `Npp8u *pDst`, `int nDstStep`, `NppiSize oSizeROI`)
Four 8-bit unsigned char channel image logical not.
- `NppStatus nppiNot_8u_C4IR` (`Npp8u *pSrcDst`, `int nSrcDstStep`, `NppiSize oSizeROI`)
Four 8-bit unsigned char channel in place image logical not.

AlphaCompC Image

Composite two images using constant alpha values.

- `NppStatus nppiAlphaCompC_8u_C1R` (`const Npp8u *pSrc1`, `int nSrc1Step`, `Npp8u nAlpha1`, `const Npp8u *pSrc2`, `int nSrc2Step`, `Npp8u nAlpha2`, `Npp8u *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `NppiAlphaOp eAlphaOp`)
One 8-bit unsigned char channel image composition using constant alpha.
- `NppStatus nppiAlphaCompC_8u_C3R` (`const Npp8u *pSrc1`, `int nSrc1Step`, `Npp8u nAlpha1`, `const Npp8u *pSrc2`, `int nSrc2Step`, `Npp8u nAlpha2`, `Npp8u *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `NppiAlphaOp eAlphaOp`)
Three 8-bit unsigned char channel image composition using constant alpha.
- `NppStatus nppiAlphaCompC_8u_C4R` (`const Npp8u *pSrc1`, `int nSrc1Step`, `Npp8u nAlpha1`, `const Npp8u *pSrc2`, `int nSrc2Step`, `Npp8u nAlpha2`, `Npp8u *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `NppiAlphaOp eAlphaOp`)
Four 8-bit unsigned char channel image composition using constant alpha.
- `NppStatus nppiAlphaCompC_8u_AC4R` (`const Npp8u *pSrc1`, `int nSrc1Step`, `Npp8u nAlpha1`, `const Npp8u *pSrc2`, `int nSrc2Step`, `Npp8u nAlpha2`, `Npp8u *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `NppiAlphaOp eAlphaOp`)
Four 8-bit unsigned char channel image composition with alpha using constant source alpha.
- `NppStatus nppiAlphaCompC_8s_C1R` (`const Npp8s *pSrc1`, `int nSrc1Step`, `Npp8s nAlpha1`, `const Npp8s *pSrc2`, `int nSrc2Step`, `Npp8s nAlpha2`, `Npp8s *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `NppiAlphaOp eAlphaOp`)
One 8-bit signed char channel image composition using constant alpha.
- `NppStatus nppiAlphaCompC_16u_C1R` (`const Npp16u *pSrc1`, `int nSrc1Step`, `Npp16u nAlpha1`, `const Npp16u *pSrc2`, `int nSrc2Step`, `Npp16u nAlpha2`, `Npp16u *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `NppiAlphaOp eAlphaOp`)
One 16-bit unsigned short channel image composition using constant alpha.
- `NppStatus nppiAlphaCompC_16u_C3R` (`const Npp16u *pSrc1`, `int nSrc1Step`, `Npp16u nAlpha1`, `const Npp16u *pSrc2`, `int nSrc2Step`, `Npp16u nAlpha2`, `Npp16u *pDst`, `int nDstStep`, `NppiSize oSizeROI`, `NppiAlphaOp eAlphaOp`)
Three 16-bit unsigned short channel image composition using constant alpha.

- `NppStatus nppiAlphaCompC_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, `Npp16u` nAlpha1, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` nAlpha2, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppiAlphaOp` eAlphaOp)

Four 16-bit unsigned short channel image composition using constant alpha.

- `NppStatus nppiAlphaCompC_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, `Npp16u` nAlpha1, const `Npp16u` *pSrc2, int nSrc2Step, `Npp16u` nAlpha2, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppiAlphaOp` eAlphaOp)

Four 16-bit unsigned short channel image composition with alpha using constant source alpha.

- `NppStatus nppiAlphaCompC_16s_C1R` (const `Npp16s` *pSrc1, int nSrc1Step, `Npp16s` nAlpha1, const `Npp16s` *pSrc2, int nSrc2Step, `Npp16s` nAlpha2, `Npp16s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppiAlphaOp` eAlphaOp)

One 16-bit signed short channel image composition using constant alpha.

- `NppStatus nppiAlphaCompC_32u_C1R` (const `Npp32u` *pSrc1, int nSrc1Step, `Npp32u` nAlpha1, const `Npp32u` *pSrc2, int nSrc2Step, `Npp32u` nAlpha2, `Npp32u` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppiAlphaOp` eAlphaOp)

One 32-bit unsigned integer channel image composition using constant alpha.

- `NppStatus nppiAlphaCompC_32s_C1R` (const `Npp32s` *pSrc1, int nSrc1Step, `Npp32s` nAlpha1, const `Npp32s` *pSrc2, int nSrc2Step, `Npp32s` nAlpha2, `Npp32s` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppiAlphaOp` eAlphaOp)

One 32-bit signed integer channel image composition using constant alpha.

- `NppStatus nppiAlphaCompC_32f_C1R` (const `Npp32f` *pSrc1, int nSrc1Step, `Npp32f` nAlpha1, const `Npp32f` *pSrc2, int nSrc2Step, `Npp32f` nAlpha2, `Npp32f` *pDst, int nDstStep, `NppiSize` oSizeROI, `NppiAlphaOp` eAlphaOp)

One 32-bit floating point channel image composition using constant alpha.

AlphaPremulC Image

Premultiplies pixels of an image using a constant alpha value.

- `NppStatus nppiAlphaPremulC_8u_C1R` (const `Npp8u` *pSrc1, int nSrc1Step, `Npp8u` nAlpha1, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_8u_C1IR` (`Npp8u` nAlpha1, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 8-bit unsigned char channel in place image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_8u_C3R` (const `Npp8u` *pSrc1, int nSrc1Step, `Npp8u` nAlpha1, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 8-bit unsigned char channel image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_8u_C3IR` (`Npp8u` nAlpha1, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 8-bit unsigned char channel in place image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_8u_C4R` (const `Npp8u` *pSrc1, int nSrc1Step, `Npp8u` nAlpha1, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_8u_C4IR` (`Npp8u` nAlpha1, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_8u_AC4R` (const `Npp8u` *pSrc1, int nSrc1Step, `Npp8u` nAlpha1, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel image premultiplication with alpha using constant alpha.

- `NppStatus nppiAlphaPremulC_8u_AC4IR` (`Npp8u` nAlpha1, `Npp8u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 8-bit unsigned char channel in place image premultiplication with alpha using constant alpha.

- `NppStatus nppiAlphaPremulC_16u_C1R` (const `Npp16u` *pSrc1, int nSrc1Step, `Npp16u` nAlpha1, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_16u_C1IR` (`Npp16u` nAlpha1, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

One 16-bit unsigned short channel in place image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_16u_C3R` (const `Npp16u` *pSrc1, int nSrc1Step, `Npp16u` nAlpha1, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_16u_C3IR` (`Npp16u` nAlpha1, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Three 16-bit unsigned short channel in place image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_16u_C4R` (const `Npp16u` *pSrc1, int nSrc1Step, `Npp16u` nAlpha1, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_16u_C4IR` (`Npp16u` nAlpha1, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image premultiplication using constant alpha.

- `NppStatus nppiAlphaPremulC_16u_AC4R` (const `Npp16u` *pSrc1, int nSrc1Step, `Npp16u` nAlpha1, `Npp16u` *pDst, int nDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel image premultiplication with alpha using constant alpha.

- `NppStatus nppiAlphaPremulC_16u_AC4IR` (`Npp16u` nAlpha1, `Npp16u` *pSrcDst, int nSrcDstStep, `NppiSize` oSizeROI)

Four 16-bit unsigned short channel in place image premultiplication with alpha using constant alpha.

AlphaComp Image

Composite two images using alpha opacity values contained in each image.

- **NppStatus nppiAlphaComp_8u_AC1R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)
One 8-bit unsigned char channel image composition using image alpha values (0 - max channel pixel value).
- **NppStatus nppiAlphaComp_8u_AC4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)
Four 8-bit unsigned char channel image composition using image alpha values (0 - max channel pixel value).
- **NppStatus nppiAlphaComp_8s_AC1R** (const **Npp8s** *pSrc1, int nSrc1Step, const **Npp8s** *pSrc2, int nSrc2Step, **Npp8s** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)
One 8-bit signed char channel image composition using image alpha values (0 - max channel pixel value).
- **NppStatus nppiAlphaComp_16u_AC1R** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)
One 16-bit unsigned short channel image composition using image alpha values (0 - max channel pixel value).
- **NppStatus nppiAlphaComp_16u_AC4R** (const **Npp16u** *pSrc1, int nSrc1Step, const **Npp16u** *pSrc2, int nSrc2Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)
Four 16-bit unsigned short channel image composition using image alpha values (0 - max channel pixel value).
- **NppStatus nppiAlphaComp_16s_AC1R** (const **Npp16s** *pSrc1, int nSrc1Step, const **Npp16s** *pSrc2, int nSrc2Step, **Npp16s** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)
One 16-bit signed short channel image composition using image alpha values (0 - max channel pixel value).
- **NppStatus nppiAlphaComp_32u_AC1R** (const **Npp32u** *pSrc1, int nSrc1Step, const **Npp32u** *pSrc2, int nSrc2Step, **Npp32u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)
One 32-bit unsigned integer channel image composition using image alpha values (0 - max channel pixel value).
- **NppStatus nppiAlphaComp_32u_AC4R** (const **Npp32u** *pSrc1, int nSrc1Step, const **Npp32u** *pSrc2, int nSrc2Step, **Npp32u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)
Four 32-bit unsigned integer channel image composition using image alpha values (0 - max channel pixel value).
- **NppStatus nppiAlphaComp_32s_AC1R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)
One 32-bit signed integer channel image composition using image alpha values (0 - max channel pixel value).
- **NppStatus nppiAlphaComp_32s_AC4R** (const **Npp32s** *pSrc1, int nSrc1Step, const **Npp32s** *pSrc2, int nSrc2Step, **Npp32s** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)

Four 32-bit signed integer channel image composition using image alpha values (0 - max channel pixel value).

- **NppStatus nppiAlphaComp_32f_AC1R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)

One 32-bit floating point channel image composition using image alpha values (0.0 - 1.0).

- **NppStatus nppiAlphaComp_32f_AC4R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiAlphaOp** eAlphaOp)

Four 32-bit floating point channel image composition using image alpha values (0.0 - 1.0).

AlphaPremul Image

Premultiplies image pixels by image alpha opacity values.

- **NppStatus nppiAlphaPremul_8u_AC4R** (const **Npp8u** *pSrc1, int nSrc1Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 8-bit unsigned char channel image premultiplication with pixel alpha (0 - max channel pixel value).

- **NppStatus nppiAlphaPremul_8u_AC4IR** (**Npp8u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 8-bit unsigned char channel in place image premultiplication with pixel alpha (0 - max channel pixel value).

- **NppStatus nppiAlphaPremul_16u_AC4R** (const **Npp16u** *pSrc1, int nSrc1Step, **Npp16u** *pDst, int nDstStep, **NppiSize** oSizeROI)

Four 16-bit unsigned short channel image premultiplication with pixel alpha (0 - max channel pixel value).

- **NppStatus nppiAlphaPremul_16u_AC4IR** (**Npp16u** *pSrcDst, int nSrcDstStep, **NppiSize** oSizeROI)

Four 16-bit unsigned short channel in place image premultiplication with pixel alpha (0 - max channel pixel value).

7.7.1 Function Documentation

7.7.1.1 NppStatus nppiAbs_16s_AC4IR (Npp16s *pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

Four 16-bit signed short channel in place image absolute value with unmodified alpha.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.2 NppStatus nppiAbs_16s_AC4R (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit signed short channel image absolute value with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.3 NppStatus nppiAbs_16s_C1IR (Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 16-bit signed short channel in place image absolute value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.4 NppStatus nppiAbs_16s_C1R (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 16-bit signed short channel image absolute value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.5 NppStatus nppiAbs_16s_C3IR (Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 16-bit signed short channel in place image absolute value.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.6 NppStatus nppiAbs_16s_C3R (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 16-bit signed short channel image absolute value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.7 NppStatus nppiAbs_16s_C4IR (Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit signed short channel in place image absolute value.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.8 NppStatus nppiAbs_16s_C4R (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit signed short channel image absolute value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.9 NppStatus nppiAbs_32f_AC4IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel in place image absolute value with unmodified alpha.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.10 NppStatus nppiAbs_32f_AC4R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel image absolute value with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.11 NppStatus nppiAbs_32f_C1IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel in place image absolute value.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.12 NppStatus nppiAbs_32f_C1R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel image absolute value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.13 NppStatus nppiAbs_32f_C3IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel in place image absolute value.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.14 NppStatus nppiAbs_32f_C3R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel image absolute value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.15 NppStatus nppiAbs_32f_C4IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel in place image absolute value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.16 NppStatus nppiAbs_32f_C4R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel image absolute value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.17 `NppStatus nppiAbsDiff_16u_C1R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel absolute difference of image1 minus image2.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.18 `NppStatus nppiAbsDiff_32f_C1R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel absolute difference of image1 minus image2.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.19 `NppStatus nppiAbsDiff_8u_C1R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel absolute difference of image1 minus image2.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.20 `NppStatus nppiAbsDiffC_16u_C1R (const Npp16u * pSrc1, int nSrc1Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, Npp32u nConstant)`

One 16-bit unsigned short channel image absolute difference with constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.21 `NppStatus nppiAbsDiffC_32f_C1R (const Npp32f * pSrc1, int nSrc1Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, Npp32f nConstant)`

One 32-bit floating point channel image absolute difference with constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.22 NppStatus nppiAbsDiffC_8u_C1R (const Npp8u * *pSrc1*, int *nSrc1Step*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, Npp8u *nConstant*)

One 8-bit unsigned char channel image absolute difference with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.23 NppStatus nppiAdd_16s_AC4IRSfs (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel with unmodified alpha in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.24 NppStatus nppiAdd_16s_AC4RSfs (const Npp16s * *pSrc1*, int *nSrc1Step*, const Npp16s * *pSrc2*, int *nSrc2Step*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel with unmodified alpha image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.25 NppStatus npAdd_16s_C1RSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit signed short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.26 NppStatus npAdd_16s_C1RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit signed short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.27 NppStatus npAdd_16s_C3IRSfs (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.28 NppStatus npAdd_16s_C3RSfs (const Npp16s * *pSrc1*, int *nSrc1Step*, const Npp16s * *pSrc2*, int *nSrc2Step*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.29 NppStatus npAdd_16s_C4IRSfs (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.30 NppStatus npAdd_16s_C4RSfs (const Npp16s * *pSrc1*, int *nSrc1Step*, const Npp16s * *pSrc2*, int *nSrc2Step*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.31 NppStatus npAdd_16sc_AC4IRSfs (const Npp16sc * *pSrc*, int *nSrcStep*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.32 **NppStatus nppiAdd_16sc_AC4RSfs** (const Npp16sc * *pSrc1*, int *nSrc1Step*, const Npp16sc * *pSrc2*, int *nSrc2Step*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.33 **NppStatus nppiAdd_16sc_C1IRSfs** (const Npp16sc * *pSrc*, int *nSrcStep*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.34 **NppStatus nppiAdd_16sc_C1RSfs** (const Npp16sc * *pSrc1*, int *nSrc1Step*, const Npp16sc * *pSrc2*, int *nSrc2Step*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.35 **NppStatus nppiAdd_16sc_C3IRSfs** (const Npp16sc * *pSrc*, int *nSrcStep*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image addition, scale by $2^{-(nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.36 **NppStatus nppiAdd_16sc_C3RSfs** (const Npp16sc * *pSrc1*, int *nSrc1Step*, const Npp16sc * *pSrc2*, int *nSrc2Step*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image addition, scale by $2^{-(nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.37 NppStatus nppiAdd_16u_AC4IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.38 NppStatus nppiAdd_16u_AC4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.39 NppStatus nppiAdd_16u_C1IRSfs (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit unsigned short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.40 NppStatus nppiAdd_16u_C1RSfs (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pSrc2*, int *nSrc2Step*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit unsigned short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.41 NppStatus nppiAdd_16u_C3IRSfs (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit unsigned short channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.42 NppStatus npAdd_16u_C3RSfs (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pSrc2*, int *nSrc2Step*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel image addition, scale by $2^{-(nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.43 NppStatus npAdd_16u_C4IRSfs (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit unsigned short channel in place image addition, scale by $2^{-(nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.44 NppStatus nppiAdd_16u_C4RSfs (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pSrc2*, int *nSrc2Step*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit unsigned short channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.45 NppStatus nppiAdd_32f_AC4IR (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel with unmodified alpha in place image addition.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.46 NppStatus nppiAdd_32f_AC4R (const Npp32f * *pSrc1*, int *nSrc1Step*, const Npp32f * *pSrc2*, int *nSrc2Step*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel with unmodified alpha image addition.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.47 NppStatus npAdd_32f_C1R (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

One 32-bit floating point channel in place image addition.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.48 NppStatus npAdd_32f_C1R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

One 32-bit floating point channel image addition.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.49 NppStatus nppiAdd_32f_C3IR (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel in place image addition.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.50 NppStatus nppiAdd_32f_C3R (const Npp32f * *pSrc1*, int *nSrc1Step*, const Npp32f * *pSrc2*, int *nSrc2Step*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel image addition.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.51 NppStatus nppiAdd_32f_C4IR (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel in place image addition.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.52 **NppStatus nppiAdd_32f_C4R** (const Npp32f * *pSrc1*, int *nSrc1Step*, const Npp32f * *pSrc2*, int *nSrc2Step*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel image addition.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.53 **NppStatus nppiAdd_32fc_AC4IR** (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image addition.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.54 **NppStatus nppiAdd_32fc_AC4R** (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc * *pSrc2*, int *nSrc2Step*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image addition.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.55 NppStatus nppiAdd_32fc_C1IR (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image addition.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.56 NppStatus nppiAdd_32fc_C1R (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc * *pSrc2*, int *nSrc2Step*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image addition.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.57 **NppStatus nppiAdd_32fc_C3IR** (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image addition.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.58 **NppStatus nppiAdd_32fc_C3R** (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc * *pSrc2*, int *nSrc2Step*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image addition.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.59 **NppStatus nppiAdd_32fc_C4IR** (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image addition.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.60 NppStatus nppiAdd_32fc_C4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pSrc2, int nSrc2Step, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image addition.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.61 NppStatus nppiAdd_32s_C1IRSfs (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 32-bit signed integer channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.62 NppStatus nppiAdd_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Note: This function is to be deprecated in future NPP releases, use the function above with a scale factor of 0 instead.

32-bit image add. Add the pixel values of corresponding pixels in the ROI and write them to the output image.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.63 NppStatus npAdd_32s_C1RSfs (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32s * *pSrc2*, int *nSrc2Step*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 32-bit signed integer channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.64 NppStatus npAdd_32s_C3IRSfs (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 32-bit signed integer channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.65 **NppStatus nppiAdd_32s_C3RSfs** (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32s * *pSrc2*, int *nSrc2Step*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 32-bit signed integer channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.66 **NppStatus nppiAdd_32sc_AC4IRSfs** (const Npp32sc * *pSrc*, int *nSrcStep*, Npp32sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.67 **NppStatus nppiAdd_32sc_AC4RSfs** (const Npp32sc * *pSrc1*, int *nSrc1Step*, const Npp32sc * *pSrc2*, int *nSrc2Step*, Npp32sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.68 `NppStatus nppiAdd_32sc_C1RSfs (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.69 `NppStatus nppiAdd_32sc_C1RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pSrc2, int nSrc2Step, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.70 **NppStatus nppiAdd_32sc_C3IRSfs** (const Npp32sc * *pSrc*, int *nSrcStep*, Npp32sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.71 **NppStatus nppiAdd_32sc_C3RSfs** (const Npp32sc * *pSrc1*, int *nSrc1Step*, const Npp32sc * *pSrc2*, int *nSrc2Step*, Npp32sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.72 **NppStatus nppiAdd_8u_AC4IRSfs** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel with unmodified alpha in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.73 `NppStatus npAdd_8u_AC4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.74 `NppStatus npAdd_8u_C1IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.75 `NppStatus nppiAdd_8u_C1RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.76 `NppStatus nppiAdd_8u_C3IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.77 `NppStatus nppiAdd_8u_C3RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 8-bit unsigned char channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.78 NppStatus nppiAdd_8u_C4IRSfs (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel in place image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.79 NppStatus nppiAdd_8u_C4RSfs (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pSrc2*, int *nSrc2Step*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel image addition, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.80 NppStatus nppiAddC_16s_AC4IRSfs (const Npp16s * *pConstants*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel with unmodified alpha in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.81 NppStatus nppiAddC_16s_AC4RSfs (const Npp16s * *pSrc1*, int *nSrc1Step*, const Npp16s * *pConstants*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel with unmodified alpha image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.82 NppStatus nppiAddC_16s_C1IRSfs (const Npp16s *nConstant*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short channel in place image add constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.83 `NppStatus nppiAddC_16s_C1RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s nConstant, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.84 `NppStatus nppiAddC_16s_C3IRSfs (const Npp16s * pConstants, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short channel in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.85 `NppStatus nppiAddC_16s_C3RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pConstants, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.86 NppStatus npAddC_16s_C4IRSfs (const Npp16s * *pConstants*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.87 NppStatus npAddC_16s_C4RSfs (const Npp16s * *pSrc1*, int *nSrc1Step*, const Npp16s * *pConstants*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.88 **NppStatus nppiAddC_16sc_AC4IRSfs** (const Npp16sc * *pConstants*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.89 **NppStatus nppiAddC_16sc_AC4RSfs** (const Npp16sc * *pSrc1*, int *nSrc1Step*, const Npp16sc * *pConstants*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.90 **NppStatus nppiAddC_16sc_C1IRSfs** (const Npp16sc *nConstant*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image add constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.91 NppStatus nppiAddC_16sc_C1RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc nConstant, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.92 NppStatus nppiAddC_16sc_C3IRSfs (const Npp16sc * pConstants, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.93 NppStatus nppiAddC_16sc_C3RSfs (const Npp16sc * *pSrc1*, int *nSrc1Step*, const Npp16sc * *pConstants*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image add constant, scale, then clamp to saturated value.

Parameters:

- pSrc1* [Source-Image Pointer](#).
- nSrc1Step* [Source-Image Line Step](#).
- pConstants* pointer to a list of constant values, one per channel.
- pDst* [Destination-Image Pointer](#).
- nDstStep* [Destination-Image Line Step](#).
- oSizeROI* [Region-of-Interest \(ROI\)](#).
- nScaleFactor* [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.94 NppStatus nppiAddC_16u_AC4IRSfs (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit unsigned short channel with unmodified alpha in place image add constant, scale, then clamp to saturated value.

Parameters:

- pConstants* pointer to a list of constant values, one per channel.
- pSrcDst* [In-Place Image Pointer](#).
- nSrcDstStep* [In-Place-Image Line Step](#).
- oSizeROI* [Region-of-Interest \(ROI\)](#).
- nScaleFactor* [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.95 NppStatus nppiAddC_16u_AC4RSfs (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit unsigned short channel with unmodified alpha image add constant, scale, then clamp to saturated value.

Parameters:

- pSrc1* [Source-Image Pointer](#).
- nSrc1Step* [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.96 NppStatus nppiAddC_16u_C1RSfs (const Npp16u nConstant, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image add constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.97 NppStatus nppiAddC_16u_C1RSfs (const Npp16u * pSrcI, int nSrcIStep, const Npp16u nConstant, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.98 **NppStatus nppiAddC_16u_C3IRSfs** (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.99 **NppStatus nppiAddC_16u_C3RSfs** (const Npp16u * *pSrcI*, int *nSrcIStep*, const Npp16u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrcI Source-Image Pointer.
nSrcIStep Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.100 **NppStatus nppiAddC_16u_C4IRSfs** (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit unsigned short channel in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.101 `NppStatus nppiAddC_16u_C4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.102 `NppStatus nppiAddC_32f_AC4IR (const Npp32f * pConstants, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel with unmodified alpha in place image add constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.103 `NppStatus nppiAddC_32f_AC4R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pConstants, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel with unmodified alpha image add constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.104 **NppStatus nppiAddC_32f_C1IR** (const Npp32f *nConstant*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel in place image add constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.105 **NppStatus nppiAddC_32f_C1R** (const Npp32f * *pSrcI*, int *nSrcIStep*, const Npp32f *nConstant*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel image add constant.

Parameters:

pSrcI Source-Image Pointer.
nSrcIStep Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.106 **NppStatus nppiAddC_32f_C3IR** (const Npp32f * *pConstants*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel in place image add constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.107 NppStatus nppiAddC_32f_C3R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pConstants, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Three 32-bit floating point channel image add constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.108 NppStatus nppiAddC_32f_C4IR (const Npp32f * pConstants, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

Four 32-bit floating point channel in place image add constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.109 NppStatus nppiAddC_32f_C4R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pConstants, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Four 32-bit floating point channel image add constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.110 `NppStatus nppiAddC_32fc_AC4IR (const Npp32fc * pConstants, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha in place image add constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.111 `NppStatus nppiAddC_32fc_AC4R (const Npp32fc * pSrcI, int nSrcIStep, const Npp32fc * pConstants, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha image add constant.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.112 `NppStatus nppiAddC_32fc_C1IR (const Npp32fc nConstant, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image add constant.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.113 `NppStatus nppiAddC_32fc_C1R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc nConstant, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image add constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.114 `NppStatus nppiAddC_32fc_C3IR (const Npp32fc * pConstants, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image add constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.115 `NppStatus nppiAddC_32fc_C3R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pConstants, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image add constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.116 NppStatus nppiAddC_32fc_C4IR (const Npp32fc * pConstants, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image add constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.117 NppStatus nppiAddC_32fc_C4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pConstants, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image add constant.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.118 NppStatus nppiAddC_32s_C1IRSfs (const Npp32s nConstant, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 32-bit signed integer channel in place image add constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.119 `NppStatus nppiAddC_32s_C1RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s nConstant, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.120 `NppStatus nppiAddC_32s_C3IRSfs (const Npp32s * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer channel in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.121 `NppStatus nppiAddC_32s_C3RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.122 `NppStatus nppiAddC_32sc_AC4IRSfs (const Npp32sc * pConstants, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.123 `NppStatus nppiAddC_32sc_AC4RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pConstants, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.124 NppStatus npplAddC_32sc_C1RSfs (const Npp32sc nConstant, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image add constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.125 NppStatus npplAddC_32sc_C1RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc nConstant, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.126 `NppStatus nppiAddC_32sc_C3IRSfs (const Npp32sc * pConstants, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.127 `NppStatus nppiAddC_32sc_C3RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pConstants, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.128 `NppStatus nppiAddC_8u_AC4IRSfs (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel..

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.129 `NppStatus nppiAddC_8u_AC4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel..

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.130 `NppStatus nppiAddC_8u_C1IRSfs (const Npp8u nConstant, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel in place image add constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.131 **NppStatus nppiAddC_8u_C1RSfs** (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u *nConstant*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 8-bit unsigned char channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.132 **NppStatus nppiAddC_8u_C3IRSfs** (const Npp8u * *pConstants*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 8-bit unsigned char channel 8-bit unsigned char in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel..
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.133 **NppStatus nppiAddC_8u_C3RSfs** (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pConstants*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 8-bit unsigned char channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel..
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.134 NppStatus nppiAddC_8u_C4IRSfs (const Npp8u * *pConstants*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel in place image add constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.135 NppStatus nppiAddC_8u_C4RSfs (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pConstants*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel image add constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel..

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.136 `NppStatus nppiAddProduct_16u32f_C1IMR (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, const Npp8u * pMask, int nMaskStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel image product added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.137 `NppStatus nppiAddProduct_16u32f_C1IR (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel image product added to in place floating point destination image.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.138 `NppStatus nppiAddProduct_32f_C1IMR (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, const Npp8u * pMask, int nMaskStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel image product added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.139 `NppStatus nppiAddProduct_32f_C1IR (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel image product added to in place floating point destination image.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.140 `NppStatus nppiAddProduct_8u32f_C1IMR (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, const Npp8u * pMask, int nMaskStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel image product added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.

pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.141 `NppStatus nppiAddProduct_8u32f_C1IR (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel image product added to in place floating point destination image.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.142 `NppStatus nppiAddSquare_16u32f_C1IMR (const Npp16u * pSrc, int nSrcStep, const Npp8u * pMask, int nMaskStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel image squared then added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.143 NppStatus nppiAddSquare_16u32f_C1IR (const Npp16u * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

One 16-bit unsigned short channel image squared then added to in place floating point destination image.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.144 NppStatus nppiAddSquare_32f_C1IMR (const Npp32f * pSrc, int nSrcStep, const Npp8u * pMask, int nMaskStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

One 32-bit floating point channel image squared then added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.145 NppStatus nppiAddSquare_32f_C1IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

One 32-bit floating point channel image squared then added to in place floating point destination image.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.146 NppStatus nppiAddSquare_8u32f_C1IMR (const Npp8u * *pSrc*, int *nSrcStep*, const Npp8u * *pMask*, int *nMaskStep*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel image squared then added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.147 NppStatus nppiAddSquare_8u32f_C1IR (const Npp8u * *pSrc*, int *nSrcStep*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel image squared then added to in place floating point destination image.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.148 NppStatus nppiAddWeighted_16u32f_C1IMR (const Npp16u * *pSrc*, int *nSrcStep*, const Npp8u * *pMask*, int *nMaskStep*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, Npp32f *nAlpha*)

One 16-bit unsigned short channel alpha weighted image added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pMask Mask-Image Pointer.

nMaskStep Mask-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nAlpha Alpha weight to be applied to source image pixels (0.0F to 1.0F)

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.149 NppStatus npaiAddWeighted_16u32f_C1IR (const Npp16u * *pSrc*, int *nSrcStep*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, Npp32f *nAlpha*)

One 16-bit unsigned short channel alpha weighted image added to in place floating point destination image.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nAlpha Alpha weight to be applied to source image pixels (0.0F to 1.0F)

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.150 NppStatus npaiAddWeighted_32f_C1IMR (const Npp32f * *pSrc*, int *nSrcStep*, const Npp8u * *pMask*, int *nMaskStep*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, Npp32f *nAlpha*)

One 32-bit floating point channel alpha weighted image added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nAlpha Alpha weight to be applied to source image pixels (0.0F to 1.0F)

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.151 `NppStatus nppiAddWeighted_32f_C1IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, Npp32f nAlpha)`

One 32-bit floating point channel alpha weighted image added to in place floating point destination image.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nAlpha Alpha weight to be applied to source image pixels (0.0F to 1.0F)

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.152 `NppStatus nppiAddWeighted_8u32f_C1IMR (const Npp8u * pSrc, int nSrcStep, const Npp8u * pMask, int nMaskStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, Npp32f nAlpha)`

One 8-bit unsigned char channel alpha weighted image added to in place floating point destination image using filter mask (updates destination when mask is non-zero).

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pMask Mask-Image Pointer.
nMaskStep Mask-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nAlpha Alpha weight to be applied to source image pixels (0.0F to 1.0F)

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.153 `NppStatus nppiAddWeighted_8u32f_C1IR (const Npp8u * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, Npp32f nAlpha)`

One 8-bit unsigned char channel alpha weighted image added to in place floating point destination image.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nAlpha Alpha weight to be applied to source image pixels (0.0F to 1.0F)

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.154 `NppStatus nppiAlphaComp_16s_AC1R (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 16-bit signed short channel image composition using image alpha values (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.155 `NppStatus nppiAlphaComp_16u_AC1R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 16-bit unsigned short channel image composition using image alpha values (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.156 `NppStatus nppiAlphaComp_16u_AC4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Four 16-bit unsigned short channel image composition using image alpha values (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.157 `NppStatus nppiAlphaComp_32f_AC1R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 32-bit floating point channel image composition using image alpha values (0.0 - 1.0).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.158 `NppStatus nppiAlphaComp_32f_AC4R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Four 32-bit floating point channel image composition using image alpha values (0.0 - 1.0).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.159 `NppStatus nppiAlphaComp_32s_AC1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 32-bit signed integer channel image composition using image alpha values (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.160 `NppStatus nppiAlphaComp_32s_AC4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Four 32-bit signed integer channel image composition using image alpha values (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.161 `NppStatus nppiAlphaComp_32u_AC1R (const Npp32u * pSrc1, int nSrc1Step, const Npp32u * pSrc2, int nSrc2Step, Npp32u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 32-bit unsigned integer channel image composition using image alpha values (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.162 `NppStatus nppiAlphaComp_32u_AC4R (const Npp32u * pSrc1, int nSrc1Step, const Npp32u * pSrc2, int nSrc2Step, Npp32u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Four 32-bit unsigned integer channel image composition using image alpha values (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.163 `NppStatus nppiAlphaComp_8s_AC1R (const Npp8s * pSrc1, int nSrc1Step, const Npp8s * pSrc2, int nSrc2Step, Npp8s * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 8-bit signed char channel image composition using image alpha values (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.164 `NppStatus nppiAlphaComp_8u_AC1R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 8-bit unsigned char channel image composition using image alpha values (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.165 `NppStatus nppiAlphaComp_8u_AC4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Four 8-bit unsigned char channel image composition using image alpha values (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.166 `NppStatus nppiAlphaCompC_16s_C1R (const Npp16s * pSrc1, int nSrc1Step, Npp16s nAlpha1, const Npp16s * pSrc2, int nSrc2Step, Npp16s nAlpha2, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 16-bit signed short channel image composition using constant alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eAlphaOp alpha-blending operation..

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.167 `NppStatus nppiAlphaCompC_16u_AC4R (const Npp16u * pSrc1, int nSrc1Step, Npp16u nAlpha1, const Npp16u * pSrc2, int nSrc2Step, Npp16u nAlpha2, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Four 16-bit unsigned short channel image composition with alpha using constant source alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.168 `NppStatus nppiAlphaCompC_16u_C1R (const Npp16u * pSrc1, int nSrc1Step, Npp16u nAlpha1, const Npp16u * pSrc2, int nSrc2Step, Npp16u nAlpha2, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 16-bit unsigned short channel image composition using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.169 `NppStatus nppiAlphaCompC_16u_C3R (const Npp16u * pSrc1, int nSrc1Step, Npp16u nAlpha1, const Npp16u * pSrc2, int nSrc2Step, Npp16u nAlpha2, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Three 16-bit unsigned short channel image composition using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.170 `NppStatus nppiAlphaCompC_16u_C4R (const Npp16u * pSrc1, int nSrc1Step, Npp16u nAlpha1, const Npp16u * pSrc2, int nSrc2Step, Npp16u nAlpha2, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Four 16-bit unsigned short channel image composition using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.171 `NppStatus nppiAlphaCompC_32f_C1R (const Npp32f * pSrc1, int nSrc1Step, Npp32f nAlpha1, const Npp32f * pSrc2, int nSrc2Step, Npp32f nAlpha2, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 32-bit floating point channel image composition using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0.0 - 1.0).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0.0 - 1.0).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.172 `NppStatus nppiAlphaCompC_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, Npp32s nAlpha1, const Npp32s * pSrc2, int nSrc2Step, Npp32s nAlpha2, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 32-bit signed integer channel image composition using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.173 `NppStatus nppiAlphaCompC_32u_C1R (const Npp32u * pSrc1, int nSrc1Step, Npp32u nAlpha1, const Npp32u * pSrc2, int nSrc2Step, Npp32u nAlpha2, Npp32u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 32-bit unsigned integer channel image composition using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.174 `NppStatus nppiAlphaCompC_8s_C1R (const Npp8s * pSrc1, int nSrc1Step, Npp8s nAlpha1, const Npp8s * pSrc2, int nSrc2Step, Npp8s nAlpha2, Npp8s * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 8-bit signed char channel image composition using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.175 `NppStatus nppiAlphaCompC_8u_AC4R (const Npp8u * pSrc1, int nSrc1Step, Npp8u nAlpha1, const Npp8u * pSrc2, int nSrc2Step, Npp8u nAlpha2, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Four 8-bit unsigned char channel image composition with alpha using constant source alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.176 `NppStatus nppiAlphaCompC_8u_C1R (const Npp8u * pSrc1, int nSrc1Step, Npp8u nAlpha1, const Npp8u * pSrc2, int nSrc2Step, Npp8u nAlpha2, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

One 8-bit unsigned char channel image composition using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.177 `NppStatus nppiAlphaCompC_8u_C3R (const Npp8u * pSrc1, int nSrc1Step, Npp8u nAlpha1, const Npp8u * pSrc2, int nSrc2Step, Npp8u nAlpha2, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Three 8-bit unsigned char channel image composition using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.178 `NppStatus nppiAlphaCompC_8u_C4R (const Npp8u * pSrc1, int nSrc1Step, Npp8u nAlpha1, const Npp8u * pSrc2, int nSrc2Step, Npp8u nAlpha2, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppiAlphaOp eAlphaOp)`

Four 8-bit unsigned char channel image composition using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
nAlpha2 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
eAlphaOp alpha-blending operation..

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.179 NppStatus nppiAlphaPremul_16u_AC4IR (Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image premultiplication with pixel alpha (0 - max channel pixel value).

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.180 NppStatus nppiAlphaPremul_16u_AC4R (const Npp16u * *pSrc1*, int *nSrc1Step*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel image premultiplication with pixel alpha (0 - max channel pixel value).

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.181 NppStatus nppiAlphaPremul_8u_AC4IR (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel in place image premultiplication with pixel alpha (0 - max channel pixel value).

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.182 **NppStatus nppiAlphaPremul_8u_AC4R** (const Npp8u * *pSrc1*, int *nSrc1Step*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel image premultiplication with pixel alpha (0 - max channel pixel value).

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.183 **NppStatus nppiAlphaPremulC_16u_AC4IR** (Npp16u *nAlpha1*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image premultiplication with alpha using constant alpha.

Parameters:

nAlpha1 [Image alpha opacity \(0 - max channel pixel value\)](#).
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.184 **NppStatus nppiAlphaPremulC_16u_AC4R** (const Npp16u * *pSrc1*, int *nSrc1Step*, Npp16u *nAlpha1*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel image premultiplication with alpha using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 [Image alpha opacity \(0 - max channel pixel value\)](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.185 NppStatus nppiAlphaPremulC_16u_C1IR (Npp16u *nAlpha1*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 16-bit unsigned short channel in place image premultiplication using constant alpha.

Parameters:

nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.186 NppStatus nppiAlphaPremulC_16u_C1R (const Npp16u * *pSrc1*, int *nSrc1Step*, Npp16u *nAlpha1*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 16-bit unsigned short channel image premultiplication using constant alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.187 NppStatus nppiAlphaPremulC_16u_C3IR (Npp16u *nAlpha1*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 16-bit unsigned short channel in place image premultiplication using constant alpha.

Parameters:

nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.188 NppStatus nppiAlphaPremulC_16u_C3R (const Npp16u * *pSrc1*, int *nSrc1Step*, Npp16u *nAlpha1*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 16-bit unsigned short channel image premultiplication using constant alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.189 NppStatus nppiAlphaPremulC_16u_C4IR (Npp16u *nAlpha1*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image premultiplication using constant alpha.

Parameters:

nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.190 NppStatus nppiAlphaPremulC_16u_C4R (const Npp16u * *pSrc1*, int *nSrc1Step*, Npp16u *nAlpha1*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel image premultiplication using constant alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.191 NppStatus nppiAlphaPremulC_8u_AC4IR (Npp8u *nAlpha1*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel in place image premultiplication with alpha using constant alpha.

Parameters:

nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.192 NppStatus nppiAlphaPremulC_8u_AC4R (const Npp8u * *pSrc1*, int *nSrc1Step*, Npp8u *nAlpha1*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel image premultiplication with alpha using constant alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.193 NppStatus nppiAlphaPremulC_8u_C1IR (Npp8u *nAlpha1*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel in place image premultiplication using constant alpha.

Parameters:

nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.194 NppStatus nppiAlphaPremulC_8u_C1R (const Npp8u * *pSrc1*, int *nSrc1Step*, Npp8u *nAlpha1*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel image premultiplication using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.195 NppStatus nppiAlphaPremulC_8u_C3IR (Npp8u *nAlpha1*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 8-bit unsigned char channel in place image premultiplication using constant alpha.

Parameters:

nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.196 NppStatus nppiAlphaPremulC_8u_C3R (const Npp8u * *pSrc1*, int *nSrc1Step*, Npp8u *nAlpha1*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 8-bit unsigned char channel image premultiplication using constant alpha.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.197 NppStatus nppiAlphaPremulC_8u_C4IR (Npp8u *nAlpha1*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel in place image premultiplication using constant alpha.

Parameters:

nAlpha1 Image alpha opacity (0 - max channel pixel value).
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.198 NppStatus nppiAlphaPremulC_8u_C4R (const Npp8u * *pSrc1*, int *nSrc1Step*, Npp8u *nAlpha1*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel image premultiplication using constant alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nAlpha1 Image alpha opacity (0 - max channel pixel value).
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.199 NppStatus nppiAnd_16u_AC4IR (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image logical and with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.200 `NppStatus nppiAnd_16u_AC4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image logical and with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.201 `NppStatus nppiAnd_16u_C1IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel in place image logical and.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.202 `NppStatus nppiAnd_16u_C1R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel image logical and.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.203 NppStatus nppiAnd_16u_C3IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

Three 16-bit unsigned short channel in place image logical and.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.204 NppStatus nppiAnd_16u_C3R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)

Three 16-bit unsigned short channel image logical and.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.205 NppStatus nppiAnd_16u_C4IR (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image logical and.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.206 NppStatus nppiAnd_16u_C4R (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pSrc2*, int *nSrc2Step*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel image logical and.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.207 NppStatus nppiAnd_32s_AC4IR (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel in place image logical and with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.208 `NppStatus nppiAnd_32s_AC4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image logical and with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.209 `NppStatus nppiAnd_32s_C1IR (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit signed integer channel in place image logical and.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.210 `NppStatus nppiAnd_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit signed integer channel image logical and.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.211 `NppStatus nppiAnd_32s_C3IR (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel in place image logical and.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.212 `NppStatus nppiAnd_32s_C3R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel image logical and.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.213 NppStatus nppiAnd_32s_C4IR (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel in place image logical and.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.214 NppStatus nppiAnd_32s_C4R (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32s * *pSrc2*, int *nSrc2Step*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel image logical and.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.215 NppStatus nppiAnd_8u_AC4IR (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel in place image logical and with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.216 `NppStatus nppiAnd_8u_AC4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image logical and with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.217 `NppStatus nppiAnd_8u_C1IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel in place image logical and.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.218 `NppStatus nppiAnd_8u_C1R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel image logical and.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.219 `NppStatus nppiAnd_8u_C3IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel in place image logical and.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.220 `NppStatus nppiAnd_8u_C3R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel image logical and.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.221 NppStatus nppiAnd_8u_C4IR (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel in place image logical and.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.222 NppStatus nppiAnd_8u_C4R (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pSrc2*, int *nSrc2Step*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel image logical and.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.223 NppStatus nppiAndC_16u_AC4IR (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image logical and with constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.224 NppStatus nppiAndC_16u_AC4R (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel image logical and with constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.225 NppStatus nppiAndC_16u_C1IR (const Npp16u *nConstant*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 16-bit unsigned short channel in place image logical and with constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.226 NppStatus nppiAndC_16u_C1R (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u *nConstant*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 16-bit unsigned short channel image logical and with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.227 `NppStatus nppiAndC_16u_C3IR (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel in place image logical and with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.228 `NppStatus nppiAndC_16u_C3R (const Npp16u * pSrcI, int nSrcIStep, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel image logical and with constant.

Parameters:

pSrcI Source-Image Pointer.
nSrcIStep Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.229 `NppStatus nppiAndC_16u_C4IR (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel in place image logical and with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.230 `NppStatus nppiAndC_16u_C4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image logical and with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.231 `NppStatus nppiAndC_32s_AC4IR (const Npp32s * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel in place image logical and with constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.232 `NppStatus nppiAndC_32s_AC4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image logical and with constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.233 **NppStatus nppiAndC_32s_C1IR** (const Npp32s *nConstant*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit signed integer channel in place image logical and with constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.234 **NppStatus nppiAndC_32s_C1R** (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32s *nConstant*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit signed integer channel image logical and with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.235 **NppStatus nppiAndC_32s_C3IR** (const Npp32s * *pConstants*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit signed integer channel in place image logical and with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.236 NppStatus nppiAndC_32s_C3R (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32s * *pConstants*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit signed integer channel image logical and with constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.237 NppStatus nppiAndC_32s_C4IR (const Npp32s * *pConstants*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel in place image logical and with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.238 NppStatus nppiAndC_32s_C4R (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32s * *pConstants*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel image logical and with constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.239 NppStatus nppiAndC_8u_AC4IR (const Npp8u * *pConstants*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel in place image logical and with constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.240 NppStatus nppiAndC_8u_AC4R (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pConstants*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel image logical and with constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.241 NppStatus nppiAndC_8u_C1IR (const Npp8u *nConstant*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel in place image logical and with constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.242 NppStatus nppiAndC_8u_C1R (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u *nConstant*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel image logical and with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.243 NppStatus nppiAndC_8u_C3IR (const Npp8u * *pConstants*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 8-bit unsigned char channel in place image logical and with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.244 NppStatus nppiAndC_8u_C3R (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pConstants*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 8-bit unsigned char channel image logical and with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.245 `NppStatus nppiAndC_8u_C4IR (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image logical and with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.246 `NppStatus nppiAndC_8u_C4R (const Npp8u * pSrcI, int nSrcIStep, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image logical and with constant.

Parameters:

pSrcI Source-Image Pointer.
nSrcIStep Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.247 `NppStatus nppiDiv_16s_AC4IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel with unmodified alpha in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.248 `NppStatus nppiDiv_16s_AC4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel with unmodified alpha image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.249 `NppStatus nppiDiv_16s_C1IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.250 `NppStatus nppiDiv_16s_C1RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.251 `NppStatus nppiDiv_16s_C3IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.252 `NppStatus nppiDiv_16s_C3RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.253 **NppStatus nppiDiv_16s_C4IRSfs** (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.254 **NppStatus nppiDiv_16s_C4RSfs** (const Npp16s * *pSrc1*, int *nSrc1Step*, const Npp16s * *pSrc2*, int *nSrc2Step*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.255 **NppStatus nppiDiv_16sc_AC4IRSfs** (const Npp16sc * *pSrc*, int *nSrcStep*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.256 `NppStatus nppiDiv_16sc_AC4RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pSrc2, int nSrc2Step, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.257 `NppStatus nppiDiv_16sc_C1IRSfs (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.258 `NppStatus nppiDiv_16sc_C1RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pSrc2, int nSrc2Step, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.259 `NppStatus nppiDiv_16sc_C3RSfs (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.260 `NppStatus nppiDiv_16sc_C3RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pSrc2, int nSrc2Step, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.261 `NppStatus nppiDiv_16u_AC4IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel with unmodified alpha in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.262 `NppStatus nppiDiv_16u_AC4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel with unmodified alpha image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.263 NppStatus nppiDiv_16u_C1RSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.264 NppStatus nppiDiv_16u_C1RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.265 `NppStatus nppiDiv_16u_C3IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.266 `NppStatus nppiDiv_16u_C3RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.267 `NppStatus nppiDiv_16u_C4IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.268 `NppStatus nppiDiv_16u_C4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.269 `NppStatus nppiDiv_32f_AC4IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel with unmodified alpha in place image division.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.270 `NppStatus nppiDiv_32f_AC4R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel with unmodified alpha image division.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.271 `NppStatus nppiDiv_32f_C1IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel in place image division.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.272 `NppStatus nppiDiv_32f_C1R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel image division.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.273 `NppStatus nppiDiv_32f_C3IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel in place image division.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.274 `NppStatus nppiDiv_32f_C3R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit floating point channel image division.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.275 `NppStatus nppiDiv_32f_C4IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel in place image division.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.276 `NppStatus nppiDiv_32f_C4R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel image division.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.277 `NppStatus nppiDiv_32fc_AC4IR (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image division.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.278 NppStatus nppiDiv_32fc_AC4R (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc * *pSrc2*, int *nSrc2Step*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image division.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.279 NppStatus nppiDiv_32fc_C1IR (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image division.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.280 NppStatus nppiDiv_32fc_C1R (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc * *pSrc2*, int *nSrc2Step*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image division.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.281 NppStatus nppiDiv_32fc_C3IR (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image division.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.282 NppStatus nppiDiv_32fc_C3R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pSrc2, int nSrc2Step, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)

Three 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image division.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.283 NppStatus nppiDiv_32fc_C4IR (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image division.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.284 NppStatus nppiDiv_32fc_C4R (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc * *pSrc2*, int *nSrc2Step*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image division.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.285 NppStatus nppiDiv_32s_C1IRSfs (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 32-bit signed integer channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.286 `NppStatus nppiDiv_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Note: This function is to be deprecated in future NPP releases, use the function above with a scale factor of 0 instead.

32-bit image division. Divide pixels in pSrc2 by pSrc1's pixels.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.287 `NppStatus nppiDiv_32s_C1RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pSrc2 [Source-Image Pointer](#).
nSrc2Step [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.288 NppStatus nppiDiv_32s_C3IRSfs (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 32-bit signed integer channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.289 NppStatus nppiDiv_32s_C3RSfs (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32s * *pSrc2*, int *nSrc2Step*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 32-bit signed integer channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.290 NppStatus nppiDiv_32sc_AC4IRSfs (const Npp32sc * *pSrc*, int *nSrcStep*, Npp32sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.291 `NppStatus nppiDiv_32sc_AC4RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pSrc2, int nSrc2Step, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.292 `NppStatus nppiDiv_32sc_C1IRSfs (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.293 `NppStatus nppiDiv_32sc_C1RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pSrc2, int nSrc2Step, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.294 `NppStatus nppiDiv_32sc_C3RSfs (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.295 `NppStatus nppiDiv_32sc_C3RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pSrc2, int nSrc2Step, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.296 `NppStatus nppiDiv_8u_AC4IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.297 `NppStatus nppiDiv_8u_AC4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.298 NppStatus nppiDiv_8u_C1RSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.299 NppStatus nppiDiv_8u_C1RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.300 `NppStatus nppiDiv_8u_C3IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.301 `NppStatus nppiDiv_8u_C3RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.302 `NppStatus nppiDiv_8u_C4IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.303 `NppStatus npDiv_8u_C4RSfs (const Npp8u *pSrc1, int nSrc1Step, const Npp8u *pSrc2, int nSrc2Step, Npp8u *pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.304 `NppStatus npDiv_Round_16s_AC4IRSfs (const Npp16s *pSrc, int nSrcStep, Npp16s *pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 16-bit signed short channel in place image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.305 `NppStatus nppiDiv_Round_16s_AC4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 16-bit signed short channel image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.306 `NppStatus nppiDiv_Round_16s_C1IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

One 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.307 `NppStatus nppiDiv_Round_16s_C1RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

One 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.308 `NppStatus nppiDiv_Round_16s_C3IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Three 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.309 `NppStatus nppiDiv_Round_16s_C3RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Three 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pSrc2 [Source-Image Pointer](#).

nSrc2Step [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.310 `NppStatus nppiDiv_Round_16s_C4IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 16-bit signed short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.311 `NppStatus nppiDiv_Round_16s_C4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 16-bit signed short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.312 `NppStatus nppiDiv_Round_16u_AC4IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 16-bit unsigned short channel in place image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.313 `NppStatus nppiDiv_Round_16u_AC4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 16-bit unsigned short channel image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pSrc2 [Source-Image Pointer](#).

nSrc2Step [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.314 `NppStatus nppiDiv_Round_16u_C1IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

One 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.315 `NppStatus nppiDiv_Round_16u_C1RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

One 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.316 `NppStatus nppiDiv_Round_16u_C3IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Three 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.317 `NppStatus nppiDiv_Round_16u_C3RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Three 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.318 `NppStatus nppiDiv_Round_16u_C4IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 16-bit unsigned short channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.319 `NppStatus nppiDiv_Round_16u_C4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 16-bit unsigned short channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.320 `NppStatus nppiDiv_Round_8u_AC4IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 8-bit unsigned char channel in place image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.321 `NppStatus nppiDiv_Round_8u_AC4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 8-bit unsigned char channel image division with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pSrc2 [Source-Image Pointer](#).

nSrc2Step [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.322 `NppStatus nppiDiv_Round_8u_C1IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

One 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.323 `NppStatus nppiDiv_Round_8u_C1RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

One 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.324 `NppStatus nppiDiv_Round_8u_C3IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Three 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.325 `NppStatus nppiDiv_Round_8u_C3RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Three 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pSrc2 [Source-Image Pointer](#).

nSrc2Step [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.326 `NppStatus nppiDiv_Round_8u_C4IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 8-bit unsigned char channel in place image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.327 `NppStatus nppiDiv_Round_8u_C4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, NppRoundMode rndMode, int nScaleFactor)`

Four 8-bit unsigned char channel image division, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pSrc2 [Source-Image Pointer](#).

nSrc2Step [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

rndMode Result Rounding mode to be used (NPP_RND_ZERO, NPP_RND_NEAR, or NP_RND_FINANCIAL)

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.328 `NppStatus nppiDivC_16s_AC4IRSfs (const Npp16s * pConstants, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel with unmodified alpha in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.329 `NppStatus nppiDivC_16s_AC4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pConstants, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel with unmodified alpha image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.330 **NppStatus nppiDivC_16s_C1RSfs (const Npp16s nConstant, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)**

One 16-bit signed short channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.331 **NppStatus nppiDivC_16s_C1RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s nConstant, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)**

One 16-bit signed short channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.332 NppStatus nppiDivC_16s_C3IRSfs (const Npp16s * *pConstants*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.333 NppStatus nppiDivC_16s_C3RSfs (const Npp16s * *pSrc1*, int *nSrc1Step*, const Npp16s * *pConstants*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.334 NppStatus nppiDivC_16s_C4IRSfs (const Npp16s * *pConstants*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.335 `NppStatus nppiDivC_16s_C4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pConstants, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.336 `NppStatus nppiDivC_16sc_AC4IRSfs (const Npp16sc * pConstants, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.337 `NppStatus nppiDivC_16sc_AC4RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pConstants, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.338 NppStatus nppiDivC_16sc_C1IRSfs (const Npp16sc nConstant, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.339 NppStatus nppiDivC_16sc_C1RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc nConstant, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.340 **NppStatus nppiDivC_16sc_C3IRSfs** (const Npp16sc * *pConstants*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.341 **NppStatus nppiDivC_16sc_C3RSfs** (const Npp16sc * *pSrc1*, int *nSrc1Step*, const Npp16sc * *pConstants*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.342 **NppStatus nppiDivC_16u_AC4IRSfs** (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit unsigned short channel with unmodified alpha in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.343 `NppStatus nppiDivC_16u_AC4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel with unmodified alpha image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.344 `NppStatus nppiDivC_16u_C1IRSfs (const Npp16u nConstant, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit unsigned short channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.345 `NppStatus nppiDivC_16u_C1RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u nConstant, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit unsigned short channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.346 `NppStatus nppiDivC_16u_C3IRSfs (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit unsigned short channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.347 `NppStatus nppiDivC_16u_C3RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit unsigned short channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.348 NppStatus nppiDivC_16u_C4IRSfs (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.349 NppStatus nppiDivC_16u_C4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.350 **NppStatus nppiDivC_32f_AC4IR** (const Npp32f * *pConstants*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel with unmodified alpha in place image divided by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.351 **NppStatus nppiDivC_32f_AC4R** (const Npp32f * *pSrc1*, int *nSrc1Step*, const Npp32f * *pConstants*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel with unmodified alpha image divided by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.352 **NppStatus nppiDivC_32f_C1IR** (const Npp32f *nConstant*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel in place image divided by constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.353 **NppStatus nppiDivC_32f_C1R** (const Npp32f * *pSrc1*, int *nSrc1Step*, const Npp32f *nConstant*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel image divided by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.354 **NppStatus nppiDivC_32f_C3IR** (const Npp32f * *pConstants*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel in place image divided by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.355 **NppStatus nppiDivC_32f_C3R** (const Npp32f * *pSrc1*, int *nSrc1Step*, const Npp32f * *pConstants*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel image divided by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.356 **NppStatus nppiDivC_32f_C4IR** (const Npp32f * *pConstants*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel in place image divided by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.357 **NppStatus nppiDivC_32f_C4R** (const Npp32f * *pSrc1*, int *nSrc1Step*, const Npp32f * *pConstants*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel image divided by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.358 **NppStatus nppiDivC_32fc_AC4IR** (const Npp32fc * *pConstants*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha in place image divided by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.359 `NppStatus nppiDivC_32fc_AC4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pConstants, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha image divided by constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.360 `NppStatus nppiDivC_32fc_C1IR (const Npp32fc nConstant, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image divided by constant.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.361 `NppStatus nppiDivC_32fc_C1R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc nConstant, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image divided by constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.362 `NppStatus nppiDivC_32fc_C3IR (const Npp32fc * pConstants, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image divided by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.363 `NppStatus nppiDivC_32fc_C3R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pConstants, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image divided by constant.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.364 `NppStatus nppiDivC_32fc_C4IR (const Npp32fc * pConstants, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image divided by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.365 `NppStatus nppiDivC_32fc_C4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pConstants, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image divided by constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.366 `NppStatus nppiDivC_32s_C1IRSfs (const Npp32s nConstant, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.367 `NppStatus nppiDivC_32s_C1RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * nConstant, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.368 `NppStatus nppiDivC_32s_C3IRSfs (const Npp32s * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.369 `NppStatus nppiDivC_32s_C3RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.370 `NppStatus nppiDivC_32sc_AC4IRSfs (const Npp32sc * pConstants, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.371 `NppStatus nppiDivC_32sc_AC4RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pConstants, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.372 `NppStatus nppiDivC_32sc_C1IRSfs (const Npp32sc nConstant, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.373 `NppStatus nppiDivC_32sc_C1RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc nConstant, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.374 `NppStatus nppiDivC_32sc_C3IRSfs (const Npp32sc * pConstants, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.375 `NppStatus npDivC_32sc_C3RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pConstants, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.376 `NppStatus npDivC_8u_AC4IRSfs (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.377 `NppStatus nppiDivC_8u_AC4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.378 `NppStatus nppiDivC_8u_C1IRSfs (const Npp8u nConstant, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

nConstant [Constant](#).
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.379 `NppStatus nppiDivC_8u_C1RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u nConstant, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
nConstant [Constant](#).
pDst [Destination-Image Pointer](#).

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.380 NppStatus nppiDivC_8u_C3IRSfs (const Npp8u * *pConstants*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 8-bit unsigned char channel 8-bit unsigned char in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.381 NppStatus nppiDivC_8u_C3RSfs (const Npp8u * *pSrcI*, int *nSrcIStep*, const Npp8u * *pConstants*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 8-bit unsigned char channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.382 `NppStatus nppiDivC_8u_C4IRSfs (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel in place image divided by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.383 `NppStatus nppiDivC_8u_C4RSfs (const Npp8u * pSrcI, int nSrcIStep, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel image divided by constant, scale, then clamp to saturated value.

Parameters:

pSrcI [Source-Image Pointer](#).
nSrcIStep [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.384 `NppStatus nppiExp_16s_C1IRSfs (Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.385 NppStatus nppiExp_16s_C1RSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit signed short channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.386 NppStatus nppiExp_16s_C3IRSfs (Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Three 16-bit signed short channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.387 NppStatus nppiExp_16s_C3RSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

Three 16-bit signed short channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.388 NppStatus nppiExp_16u_C1IRSfs (Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit unsigned short channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.389 NppStatus nppiExp_16u_C1RSfs (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit unsigned short channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.390 NppStatus nppiExp_16u_C3IRSfs (Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.391 `NppStatus nppiExp_16u_C3RSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit unsigned short channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.392 `NppStatus nppiExp_32f_C1IR (Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel in place image exponential.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.393 `NppStatus nppiExp_32f_C1R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel image exponential.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.394 NppStatus nppiExp_32f_C3IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel in place image exponential.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.395 NppStatus nppiExp_32f_C3R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel image exponential.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.396 NppStatus nppiExp_8u_C1IRSfs (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 8-bit unsigned char channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.397 NppStatus nppiExp_8u_C1RSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 8-bit unsigned char channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.398 NppStatus nppiExp_8u_C3IRSfs (Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Three 8-bit unsigned char channel in place image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.399 NppStatus nppiExp_8u_C3RSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

Three 8-bit unsigned char channel image exponential, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.400 NppStatus nppiLn_16s_C1IRSfs (Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.401 NppStatus nppiLn_16s_C1RSfs (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.402 NppStatus nppiLn_16s_C3IRSfs (Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.403 NppStatus nppiLn_16s_C3RSfs (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.404 NppStatus nppiLn_16u_C1IRSfs (Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit unsigned short channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.405 NppStatus nppiLn_16u_C1RSfs (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit unsigned short channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.406 NppStatus nppiLn_16u_C3IRSfs (Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.407 NppStatus nppiLn_16u_C3RSfs (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.408 NppStatus nppiLn_32f_C1IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel in place image natural logarithm.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.409 NppStatus nppiLn_32f_C1R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel image natural logarithm.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.410 NppStatus nppiLn_32f_C3IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel in place image natural logarithm.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.411 NppStatus nppiLn_32f_C3R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel image natural logarithm.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.412 **NppStatus nppiLn_8u_C1IRSfs** (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 8-bit unsigned char channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.413 **NppStatus nppiLn_8u_C1RSfs** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 8-bit unsigned char channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.414 **NppStatus nppiLn_8u_C3IRSfs** (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 8-bit unsigned char channel in place image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.415 NppStatus nppiLn_8u_C3RSfs (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 8-bit unsigned char channel image natural logarithm, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.416 NppStatus nppiLShiftC_16u_AC4IR (const Npp32u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image left shift by constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.417 NppStatus nppiLShiftC_16u_AC4R (const Npp16u * *pSrcI*, int *nSrcIStep*, const Npp32u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel image left shift by constant with unmodified alpha.

Parameters:

pSrcI Source-Image Pointer.
nSrcIStep Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.418 NppStatus nppiLShiftC_16u_C1IR (const Npp32u *nConstant*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 16-bit unsigned short channel in place image left shift by constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.419 NppStatus nppiLShiftC_16u_C1R (const Npp16u * *pSrcI*, int *nSrcIStep*, const Npp32u *nConstant*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 16-bit unsigned short channel image left shift by constant.

Parameters:

pSrcI Source-Image Pointer.
nSrcIStep Source-Image Line Step.
nConstant Constant
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.420 NppStatus nppiLShiftC_16u_C3IR (const Npp32u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 16-bit unsigned short channel in place image left shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.421 `NppStatus nppiLShiftC_16u_C3R (const Npp16u * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel image left shift by constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.422 `NppStatus nppiLShiftC_16u_C4IR (const Npp32u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel in place image left shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.423 `NppStatus nppiLShiftC_16u_C4R (const Npp16u * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image left shift by constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.424 NppStatus nppiLShiftC_32s_AC4IR (const Npp32u * *pConstants*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel in place image left shift by constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.425 NppStatus nppiLShiftC_32s_AC4R (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32u * *pConstants*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel image left shift by constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.426 NppStatus nppiLShiftC_32s_C1IR (const Npp32u *nConstant*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit signed integer channel in place image left shift by constant.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.427 `NppStatus nppiLShiftC_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32u nConstant, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit signed integer channel image left shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.428 `NppStatus nppiLShiftC_32s_C3IR (const Npp32u * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel in place image left shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.429 `NppStatus nppiLShiftC_32s_C3R (const Npp32s * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel image left shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.430 `NppStatus nppiLShiftC_32s_C4IR (const Npp32u * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel in place image left shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.431 `NppStatus nppiLShiftC_32s_C4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image left shift by constant.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.432 `NppStatus nppiLShiftC_8u_AC4IR (const Npp32u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image left shift by constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.433 `NppStatus nppiLShiftC_8u_AC4R (const Npp8u * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image left shift by constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.434 `NppStatus nppiLShiftC_8u_C1IR (const Npp32u nConstant, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel in place image left shift by constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.435 `NppStatus nppiLShiftC_8u_C1R (const Npp8u * pSrc1, int nSrc1Step, const Npp32u nConstant, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel image left shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.436 `NppStatus nppiLShiftC_8u_C3IR (const Npp32u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel in place image left shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.437 `NppStatus nppiLShiftC_8u_C3R (const Npp8u * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel image left shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.438 `NppStatus nppiLShiftC_8u_C4IR (const Npp32u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image left shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.439 `NppStatus nppiLShiftC_8u_C4R (const Npp8u * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image left shift by constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.440 `NppStatus nppiMul_16s_AC4IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel with unmodified alpha in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.441 `NppStatus nppiMul_16s_AC4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel with unmodified alpha image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.442 `NppStatus nppiMul_16s_C1RSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.443 `NppStatus nppiMul_16s_C1RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.444 `NppStatus nppiMul_16s_C3IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.445 `NppStatus nppiMul_16s_C3RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.446 `NppStatus nppiMul_16s_C4IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.447 `NppStatus nppiMul_16s_C4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.448 `NppStatus nppiMul_16sc_AC4IRSfs (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.449 `NppStatus nppiMul_16sc_AC4RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pSrc2, int nSrc2Step, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.450 `NppStatus nppiMul_16sc_C1IRSfs (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.451 `NppStatus nppiMul_16sc_C1RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pSrc2, int nSrc2Step, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.452 `NppStatus nppiMul_16sc_C3IRSfs (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.453 `NppStatus nppiMul_16sc_C3RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pSrc2, int nSrc2Step, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.454 NppStatus nppiMul_16u_AC4IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.455 NppStatus nppiMul_16u_AC4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel with unmodified alpha image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.456 `NppStatus nppiMul_16u_C1IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit unsigned short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.457 `NppStatus nppiMul_16u_C1RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit unsigned short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.458 `NppStatus nppiMul_16u_C3IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit unsigned short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.459 `NppStatus nppiMul_16u_C3RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit unsigned short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.460 `NppStatus nppiMul_16u_C4IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.461 `NppStatus nppiMul_16u_C4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.462 `NppStatus nppiMul_32f_AC4IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel with unmodified alpha in place image multiplication.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.463 `NppStatus nppiMul_32f_AC4R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel with unmodified alpha image multiplication.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.464 NppStatus nppiMul_32f_C1R (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

One 32-bit floating point channel in place image multiplication.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.465 NppStatus nppiMul_32f_C1R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

One 32-bit floating point channel image multiplication.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.466 NppStatus nppiMul_32f_C3IR (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel in place image multiplication.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.467 NppStatus nppiMul_32f_C3R (const Npp32f * *pSrc1*, int *nSrc1Step*, const Npp32f * *pSrc2*, int *nSrc2Step*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel image multiplication.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.468 NppStatus nppiMul_32f_C4IR (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel in place image multiplication.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.469 `NppStatus nppiMul_32f_C4R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel image multiplication.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.470 `NppStatus nppiMul_32fc_AC4IR (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image multiplication.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.471 `NppStatus nppiMul_32fc_AC4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pSrc2, int nSrc2Step, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image multiplication.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.472 `NppStatus nppiMul_32fc_C1IR (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image multiplication.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.473 `NppStatus nppiMul_32fc_C1R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pSrc2, int nSrc2Step, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image multiplication.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.474 NppStatus nppiMul_32fc_C3IR (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image multiplication.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.475 NppStatus nppiMul_32fc_C3R (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc * *pSrc2*, int *nSrc2Step*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image multiplication.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.476 NppStatus nppiMul_32fc_C4IR (const Npp32fc * *pSrc*, int *nSrcStep*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image multiplication.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.477 `NppStatus nppiMul_32fc_C4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pSrc2, int nSrc2Step, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image multiplication.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.478 `NppStatus nppiMul_32s_C1IRSfs (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.479 NppStatus nppiMul_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)

Note: This function is to be deprecated in future NPP releases, use the function above with a scale factor of 0 instead.

1 channel 32-bit image multiplication. Multiply corresponding pixels in ROI.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.480 NppStatus nppiMul_32s_C1RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 32-bit signed integer channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.481 NppStatus nppiMul_32s_C3IRSfs (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 32-bit signed integer channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.482 `NppStatus nppiMul_32s_C3RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.483 `NppStatus nppiMul_32sc_AC4IRSfs (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.484 `NppStatus nppiMul_32sc_AC4RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pSrc2, int nSrc2Step, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.485 `NppStatus nppiMul_32sc_C1IRSfs (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.486 `NppStatus nppiMul_32sc_C1RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pSrc2, int nSrc2Step, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.487 `NppStatus nppiMul_32sc_C3IRSfs (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.488 `NppStatus nppiMul_32sc_C3RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pSrc2, int nSrc2Step, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.489 `NppStatus nppiMul_8u_AC4IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.490 `NppStatus nppiMul_8u_AC4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.491 `NppStatus nppiMul_8u_C1IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.492 `NppStatus nppiMul_8u_C1RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.493 `NppStatus nppiMul_8u_C3IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.494 `NppStatus nppiMul_8u_C3RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 8-bit unsigned char channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.495 `NppStatus nppiMul_8u_C4IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel in place image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.496 `NppStatus nppiMul_8u_C4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel image multiplication, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.497 `NppStatus nppiMulC_16s_AC4IRSfs (const Npp16s * pConstants, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel with unmodified alpha in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.498 `NppStatus nppiMulC_16s_AC4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pConstants, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel with unmodified alpha image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.499 NppStatus nppiMulC_16s_C1IRSfs (const Npp16s nConstant, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit signed short channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.500 NppStatus nppiMulC_16s_C1RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s nConstant, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit signed short channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.501 `NppStatus nppiMulC_16s_C3IRSfs (const Npp16s * pConstants, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.502 `NppStatus nppiMulC_16s_C3RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pConstants, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.503 `NppStatus nppiMulC_16s_C4IRSfs (const Npp16s * pConstants, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.504 `NppStatus nppiMulC_16s_C4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pConstants, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.505 `NppStatus nppiMulC_16sc_AC4IRSfs (const Npp16sc * pConstants, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.506 `NppStatus nppiMulC_16sc_AC4RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pConstants, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.507 `NppStatus nppiMulC_16sc_C1RSfs (const Npp16sc nConstant, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.508 `NppStatus nppiMulC_16sc_C1RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc nConstant, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.509 `NppStatus nppiMulC_16sc_C3IRSfs (const Npp16sc * pConstants, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.510 `NppStatus nppiMulC_16sc_C3RSfs (const Npp16sc * pSrcI, int nSrcIStep, const Npp16sc * pConstants, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.511 `NppStatus nppiMulC_16u_AC4IRSfs (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel with unmodified alpha in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.512 `NppStatus nppiMulC_16u_AC4RSfs (const Npp16u *pSrc1, int nSrc1Step, const Npp16u *pConstants, Npp16u *pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel with unmodified alpha image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.513 `NppStatus nppiMulC_16u_C1IRSfs (const Npp16u nConstant, Npp16u *pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit unsigned short channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.514 NppStatus nppiMulC_16u_C1RSfs (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u *nConstant*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit unsigned short channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.515 NppStatus nppiMulC_16u_C3IRSfs (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.516 NppStatus nppiMulC_16u_C3RSfs (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.517 NppStatus nppiMulC_16u_C4IRSfs (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit unsigned short channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.518 NppStatus nppiMulC_16u_C4RSfs (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit unsigned short channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.519 NppStatus nppiMulC_32f_AC4IR (const Npp32f * *pConstants*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel with unmodified alpha in place image multiply by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.520 NppStatus nppiMulC_32f_AC4R (const Npp32f * *pSrcI*, int *nSrcIStep*, const Npp32f * *pConstants*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel with unmodified alpha image multiply by constant.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.521 NppStatus nppiMulC_32f_C1IR (const Npp32f *nConstant*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel in place image multiply by constant.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.522 `NppStatus nppiMulC_32f_C1R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f nConstant, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel image multiply by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.523 `NppStatus nppiMulC_32f_C3IR (const Npp32f * pConstants, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 32-bit floating point channel in place image multiply by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.524 `NppStatus nppiMulC_32f_C3R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pConstants, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit floating point channel image multiply by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.525 `NppStatus nppiMulC_32f_C4IR (const Npp32f * pConstants, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel in place image multiply by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.526 `NppStatus nppiMulC_32f_C4R (const Npp32f * pSrcI, int nSrcIStep, const Npp32f * pConstants, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel image multiply by constant.

Parameters:

pSrcI Source-Image Pointer.
nSrcIStep Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.527 `NppStatus nppiMulC_32fc_AC4IR (const Npp32fc * pConstants, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha in place image multiply by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.528 **NppStatus nppiMulC_32fc_AC4R** (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc * *pConstants*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha image multiply by constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.529 **NppStatus nppiMulC_32fc_C1IR** (const Npp32fc *nConstant*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image multiply by constant.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.530 **NppStatus nppiMulC_32fc_C1R** (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc *nConstant*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image multiply by constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.531 `NppStatus nppiMulC_32fc_C3IR (const Npp32fc * pConstants, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image multiply by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.532 `NppStatus nppiMulC_32fc_C3R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pConstants, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image multiply by constant.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.533 `NppStatus nppiMulC_32fc_C4IR (const Npp32fc * pConstants, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image multiply by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.534 `NppStatus nppiMulC_32fc_C4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pConstants, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image multiply by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.535 `NppStatus nppiMulC_32s_C1IRSfs (const Npp32s nConstant, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.536 `NppStatus nppiMulC_32s_C1RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s nConstant, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.537 `NppStatus nppiMulC_32s_C3IRSfs (const Npp32s * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.538 `NppStatus nppiMulC_32s_C3RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.539 `NppStatus nppiMulC_32sc_AC4IRSfs (const Npp32sc * pConstants, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.540 `NppStatus nppiMulC_32sc_AC4RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pConstants, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.541 `NppStatus nppiMulC_32sc_C1RSfs (const Npp32sc nConstant, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.542 `NppStatus nppiMulC_32sc_C1RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc nConstant, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.543 `NppStatus nppiMulC_32sc_C3RSfs (const Npp32sc * pConstants, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.544 `NppStatus nppiMulC_32sc_C3RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pConstants, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.545 `NppStatus nppiMulC_8u_AC4IRSfs (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.546 NppStatus nppiMulC_8u_AC4RSfs (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pConstants*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel with unmodified alpha image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.547 NppStatus nppiMulC_8u_C1IRSfs (const Npp8u *nConstant*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 8-bit unsigned char channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.548 NppStatus nppiMulC_8u_C1RSfs (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u *nConstant*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 8-bit unsigned char channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.549 `NppStatus nppiMulC_8u_C3IRSfs (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 8-bit unsigned char channel 8-bit unsigned char in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.550 `NppStatus nppiMulC_8u_C3RSfs (const Npp8u * pSrcI, int nSrcIStep, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 8-bit unsigned char channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.551 `NppStatus nppiMulC_8u_C4IRSfs (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel in place image multiply by constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.552 `NppStatus nppiMulC_8u_C4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel image multiply by constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.553 `NppStatus nppiMulCScale_16u_AC4IR (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel with unmodified alpha in place image multiply by constant and scale by max bit width value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.554 NppStatus nppiMulCScale_16u_AC4R (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel with unmodified alpha image multiply by constant and scale by max bit width value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.555 NppStatus nppiMulCScale_16u_C1IR (const Npp16u *nConstant*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 16-bit unsigned short channel in place image multiply by constant and scale by max bit width value.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.556 NppStatus nppiMulCScale_16u_C1R (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u *nConstant*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 16-bit unsigned short channel image multiply by constant and scale by max bit width value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.557 NppStatus nppiMulCScale_16u_C3IR (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 16-bit unsigned short channel in place image multiply by constant and scale by max bit width value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.558 NppStatus nppiMulCScale_16u_C3R (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 16-bit unsigned short channel image multiply by constant and scale by max bit width value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.559 NppStatus nppiMulCScale_16u_C4IR (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image multiply by constant and scale by max bit width value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.560 `NppStatus nppiMulCScale_16u_C4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image multiply by constant and scale by max bit width value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.561 `NppStatus nppiMulCScale_8u_AC4IR (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel with unmodified alpha in place image multiply by constant, scale and scale by max bit width value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.562 `NppStatus nppiMulCScale_8u_AC4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel with unmodified alpha image multiply by constant and scale by max bit width value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.563 `NppStatus nppiMulCScale_8u_C1IR (const Npp8u nConstant, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel in place image multiply by constant and scale by max bit width value.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.564 `NppStatus nppiMulCScale_8u_C1R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u nConstant, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel image multiply by constant and scale by max bit width value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.565 `NppStatus nppiMulCScale_8u_C3IR (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel 8-bit unsigned char in place image multiply by constant and scale by max bit width value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.566 `NppStatus nppiMulCScale_8u_C3R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel image multiply by constant and scale by max bit width value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.567 `NppStatus nppiMulCScale_8u_C4IR (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image multiply by constant and scale by max bit width value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.568 `NppStatus nppiMulCScale_8u_C4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image multiply by constant and scale by max bit width value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.569 `NppStatus nppiMulScale_16u_AC4IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel with unmodified alpha in place image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.570 `NppStatus nppiMulScale_16u_AC4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel with unmodified alpha image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.571 `NppStatus nppiMulScale_16u_C1IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel in place image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.572 `NppStatus nppiMulScale_16u_C1R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.573 `NppStatus nppiMulScale_16u_C3IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel in place image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.574 `NppStatus nppiMulScale_16u_C3R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.575 `NppStatus nppiMulScale_16u_C4IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel in place image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.576 `NppStatus nppiMulScale_16u_C4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.577 `NppStatus nppiMulScale_8u_AC4IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel with unmodified alpha in place image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.578 `NppStatus nppiMulScale_8u_AC4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel with unmodified alpha image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.579 `NppStatus nppiMulScale_8u_C1IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel in place image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.580 `NppStatus nppiMulScale_8u_C1R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.581 `NppStatus nppiMulScale_8u_C3IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel in place image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.582 `NppStatus nppiMulScale_8u_C3R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.583 `NppStatus nppiMulScale_8u_C4IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.584 `NppStatus nppiMulScale_8u_C4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image multiplication then scale by maximum value for pixel bit width.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.585 NppStatus nppiNot_8u_AC4IR (Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

Four 8-bit unsigned char channel in place image logical not with unmodified alpha.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.586 NppStatus nppiNot_8u_AC4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)

Four 8-bit unsigned char channel image logical not with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.587 NppStatus nppiNot_8u_C1IR (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel in place image logical not.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.588 NppStatus nppiNot_8u_C1R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel image logical not.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.589 NppStatus nppiNot_8u_C3IR (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 8-bit unsigned char channel in place image logical not.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.590 NppStatus nppiNot_8u_C3R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 8-bit unsigned char channel image logical not.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.591 NppStatus nppiNot_8u_C4IR (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel in place image logical not.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.592 NppStatus nppiNot_8u_C4R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel image logical not.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.593 `NppStatus nppiOr_16u_AC4IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel in place image logical or with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.594 `NppStatus nppiOr_16u_AC4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image logical or with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.595 `NppStatus nppiOr_16u_C1IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel in place image logical or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.596 `NppStatus nppiOr_16u_C1R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel image logical or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.597 `NppStatus nppiOr_16u_C3IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel in place image logical or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.598 `NppStatus nppiOr_16u_C3R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel image logical or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.599 `NppStatus nppiOr_16u_C4IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel in place image logical or.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.600 `NppStatus nppiOr_16u_C4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image logical or.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.601 **NppStatus nppiOr_32s_AC4IR** (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel in place image logical or with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.602 **NppStatus nppiOr_32s_AC4R** (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32s * *pSrc2*, int *nSrc2Step*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel image logical or with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.603 **NppStatus nppiOr_32s_C1IR** (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit signed integer channel in place image logical or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.604 `NppStatus nppiOr_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit signed integer channel image logical or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.605 `NppStatus nppiOr_32s_C3IR (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel in place image logical or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.606 `NppStatus nppiOr_32s_C3R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel image logical or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.607 `NppStatus nppiOr_32s_C4IR (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel in place image logical or.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.608 `NppStatus nppiOr_32s_C4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image logical or.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.609 `NppStatus nppiOr_8u_AC4IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image logical or with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.610 `NppStatus nppiOr_8u_AC4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image logical or with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.611 `NppStatus nppiOr_8u_C1IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel in place image logical or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.612 `NppStatus nppiOr_8u_C1R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel image logical or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.613 `NppStatus nppiOr_8u_C3IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel in place image logical or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.614 `NppStatus nppiOr_8u_C3R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel image logical or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.615 `NppStatus nppiOr_8u_C4IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image logical or.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.616 `NppStatus nppiOr_8u_C4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image logical or.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.617 **NppStatus nppiOrC_16u_AC4IR** (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image logical or with constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.618 **NppStatus nppiOrC_16u_AC4R** (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel image logical or with constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.619 **NppStatus nppiOrC_16u_C1IR** (const Npp16u *nConstant*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 16-bit unsigned short channel in place image logical or with constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.620 `NppStatus nppiOrC_16u_C1R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u nConstant, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel image logical or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.621 `NppStatus nppiOrC_16u_C3IR (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel in place image logical or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.622 `NppStatus nppiOrC_16u_C3R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel image logical or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.623 NppStatus nppiOrC_16u_C4IR (const Npp16u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image logical or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.624 NppStatus nppiOrC_16u_C4R (const Npp16u * *pSrcI*, int *nSrcIStep*, const Npp16u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel image logical or with constant.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.625 NppStatus nppiOrC_32s_AC4IR (const Npp32s * *pConstants*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel in place image logical or with constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.626 `NppStatus nppiOrC_32s_AC4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image logical or with constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.627 `NppStatus nppiOrC_32s_C1IR (const Npp32s nConstant, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit signed integer channel in place image logical or with constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.628 `NppStatus nppiOrC_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s nConstant, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit signed integer channel image logical or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.629 `NppStatus nppiOrC_32s_C3IR (const Npp32s * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel in place image logical or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.630 `NppStatus nppiOrC_32s_C3R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel image logical or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.631 `NppStatus nppiOrC_32s_C4IR (const Npp32s * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel in place image logical or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.632 `NppStatus nppiOrC_32s_C4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image logical or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.633 `NppStatus nppiOrC_8u_AC4IR (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image logical or with constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.634 `NppStatus nppiOrC_8u_AC4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image logical or with constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.635 NppStatus nppiOrC_8u_C1IR (const Npp8u *nConstant*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel in place image logical or with constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.636 NppStatus nppiOrC_8u_C1R (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u *nConstant*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel image logical or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.637 NppStatus nppiOrC_8u_C3IR (const Npp8u * *pConstants*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 8-bit unsigned char channel in place image logical or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.638 `NppStatus nppiOrC_8u_C3R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel image logical or with constant.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.639 `NppStatus nppiOrC_8u_C4IR (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image logical or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.640 `NppStatus nppiOrC_8u_C4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image logical or with constant.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.641 `NppStatus nppiRShiftC_16s_AC4IR (const Npp32u * pConstants, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit signed short channel in place image right shift by constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.642 `NppStatus nppiRShiftC_16s_AC4R (const Npp16s * pSrcI, int nSrcIStep, const Npp32u * pConstants, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit signed short channel image right shift by constant with unmodified alpha.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.643 `NppStatus nppiRShiftC_16s_C1IR (const Npp32u nConstant, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 16-bit signed short channel in place image right shift by constant.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.644 `NppStatus nppiRShiftC_16s_C1R (const Npp16s * pSrc1, int nSrc1Step, const Npp32u nConstant, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)`

One 16-bit signed short channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.645 `NppStatus nppiRShiftC_16s_C3IR (const Npp32u * pConstants, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 16-bit signed short channel in place image right shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.646 `NppStatus nppiRShiftC_16s_C3R (const Npp16s * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp16s * pDst, int nDstStep, NppiSize oSizeROI)`

Three 16-bit signed short channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.647 NppStatus nppiRShiftC_16s_C4IR (const Npp32u * *pConstants*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit signed short channel in place image right shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.648 NppStatus nppiRShiftC_16s_C4R (const Npp16s * *pSrc1*, int *nSrc1Step*, const Npp32u * *pConstants*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit signed short channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.649 NppStatus nppiRShiftC_16u_AC4IR (const Npp32u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image right shift by constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.650 `NppStatus nppiRShiftC_16u_AC4R (const Npp16u * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image right shift by constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.651 `NppStatus nppiRShiftC_16u_C11R (const Npp32u nConstant, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel in place image right shift by constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.652 `NppStatus nppiRShiftC_16u_C1R (const Npp16u * pSrc1, int nSrc1Step, const Npp32u nConstant, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.653 NppStatus nppiRShiftC_16u_C3IR (const Npp32u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 16-bit unsigned short channel in place image right shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.654 NppStatus nppiRShiftC_16u_C3R (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp32u * *pConstants*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 16-bit unsigned short channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.655 NppStatus nppiRShiftC_16u_C4IR (const Npp32u * *pConstants*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image right shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.656 `NppStatus nppiRShiftC_16u_C4R (const Npp16u * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.657 `NppStatus nppiRShiftC_32s_AC4IR (const Npp32u * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel in place image right shift by constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.658 `NppStatus nppiRShiftC_32s_AC4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image right shift by constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.659 NppStatus nppiRShiftC_32s_C1IR (const Npp32u *nConstant*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit signed integer channel in place image right shift by constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.660 NppStatus nppiRShiftC_32s_C1R (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32u *nConstant*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit signed integer channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.661 NppStatus nppiRShiftC_32s_C3IR (const Npp32u * *pConstants*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit signed integer channel in place image right shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.662 `NppStatus nppiRShiftC_32s_C3R (const Npp32s * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.663 `NppStatus nppiRShiftC_32s_C4IR (const Npp32u * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel in place image right shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.664 `NppStatus nppiRShiftC_32s_C4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.665 NppStatus nppiRShiftC_8s_AC4IR (const Npp32u * *pConstants*, Npp8s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 8-bit signed char channel in place image right shift by constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.666 NppStatus nppiRShiftC_8s_AC4R (const Npp8s * *pSrcI*, int *nSrcIStep*, const Npp32u * *pConstants*, Npp8s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 8-bit signed char channel image right shift by constant with unmodified alpha.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.667 NppStatus nppiRShiftC_8s_C1IR (const Npp32u *nConstant*, Npp8s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 8-bit signed char channel in place image right shift by constant.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.668 `NppStatus nppiRShiftC_8s_C1R (const Npp8s * pSrc1, int nSrc1Step, const Npp32u nConstant, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)`

One 8-bit signed char channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.669 `NppStatus nppiRShiftC_8s_C3IR (const Npp32u * pConstants, Npp8s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 8-bit signed char channel in place image right shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.670 `NppStatus nppiRShiftC_8s_C3R (const Npp8s * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)`

Three 8-bit signed char channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.671 `NppStatus nppiRShiftC_8s_C4IR (const Npp32u * pConstants, Npp8s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit signed char channel in place image right shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.672 `NppStatus nppiRShiftC_8s_C4R (const Npp8s * pSrcI, int nSrcIStep, const Npp32u * pConstants, Npp8s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit signed char channel image right shift by constant.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.673 `NppStatus nppiRShiftC_8u_AC4IR (const Npp32u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image right shift by constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.674 `NppStatus nppiRShiftC_8u_AC4R (const Npp8u * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image right shift by constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.675 `NppStatus nppiRShiftC_8u_C1IR (const Npp32u nConstant, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel in place image right shift by constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.676 `NppStatus nppiRShiftC_8u_C1R (const Npp8u * pSrc1, int nSrc1Step, const Npp32u nConstant, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.677 NppStatus nppiRShiftC_8u_C3IR (const Npp32u * *pConstants*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 8-bit unsigned char channel in place image right shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.678 NppStatus nppiRShiftC_8u_C3R (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp32u * *pConstants*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 8-bit unsigned char channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.679 NppStatus nppiRShiftC_8u_C4IR (const Npp32u * *pConstants*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel in place image right shift by constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.680 `NppStatus nppiRShiftC_8u_C4R (const Npp8u * pSrc1, int nSrc1Step, const Npp32u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image right shift by constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.681 `NppStatus nppiSqr_16s_AC4IRSfs (Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel in place image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.682 `NppStatus nppiSqr_16s_AC4RSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.683 NppStatus nppiSqr_16s_C1IRSfs (Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit signed short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.684 NppStatus nppiSqr_16s_C1RSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit signed short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.685 NppStatus nppiSqr_16s_C3IRSfs (Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Three 16-bit signed short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.686 `NppStatus nppiSqr_16s_C3RSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.687 `NppStatus nppiSqr_16s_C4IRSfs (Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.688 `NppStatus nppiSqr_16s_C4RSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.689 NppStatus nppiSqr_16u_AC4IRSfs (Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel in place image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.690 NppStatus nppiSqr_16u_AC4RSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.691 NppStatus nppiSqr_16u_C1IRSfs (Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.692 NppStatus nppiSqr_16u_C1RSfs (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit unsigned short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.693 NppStatus nppiSqr_16u_C3IRSfs (Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.694 NppStatus nppiSqr_16u_C3RSfs (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.695 NppStatus nppiSqr_16u_C4IRSfs (Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit unsigned short channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.696 NppStatus nppiSqr_16u_C4RSfs (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit unsigned short channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.697 NppStatus nppiSqr_32f_AC4IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel in place image squared with unmodified alpha.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.698 NppStatus nppiSqr_32f_AC4R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel image squared with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.699 NppStatus nppiSqr_32f_C1IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel in place image squared.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.700 NppStatus nppiSqr_32f_C1R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel image squared.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.701 NppStatus nppiSqr_32f_C3IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel in place image squared.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.702 NppStatus nppiSqr_32f_C3R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel image squared.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.703 NppStatus nppiSqr_32f_C4IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel in place image squared.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.704 NppStatus nppiSqr_32f_C4R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel image squared.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.705 NppStatus nppiSqr_8u_AC4IRSfs (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel in place image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.706 NppStatus nppiSqr_8u_AC4RSfs (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel image squared with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.707 NppStatus nppiSqr_8u_C1IRSfs (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 8-bit unsigned char channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.708 NppStatus nppiSqr_8u_C1RSfs (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 8-bit unsigned char channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.709 NppStatus nppiSqr_8u_C3IRSfs (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 8-bit unsigned char channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.710 **NppStatus nppiSqr_8u_C3RSfs** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 8-bit unsigned char channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.711 **NppStatus nppiSqr_8u_C4IRSfs** (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel in place image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.712 **NppStatus nppiSqr_8u_C4RSfs** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel image squared, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.713 NppStatus nppiSqrt_16s_AC4IRSfs (Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel in place image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.714 NppStatus nppiSqrt_16s_AC4RSfs (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.715 NppStatus nppiSqrt_16s_C1IRSfs (Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.716 NppStatus nppiSqrt_16s_C1RSfs (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.717 NppStatus nppiSqrt_16s_C3IRSfs (Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.718 NppStatus nppiSqrt_16s_C3RSfs (const Npp16s * *pSrc*, int *nSrcStep*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit signed short channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.719 NppStatus nppiSqrt_16u_AC4IRSfs (Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel in place image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.720 NppStatus nppiSqrt_16u_AC4RSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit unsigned short channel image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.721 NppStatus nppiSqrt_16u_C1IRSfs (Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.722 **NppStatus nppiSqrt_16u_C1RSfs** (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit unsigned short channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.723 **NppStatus nppiSqrt_16u_C3IRSfs** (Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.724 **NppStatus nppiSqrt_16u_C3RSfs** (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 16-bit unsigned short channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.725 NppStatus nppiSqrt_32f_AC4IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel in place image square root with unmodified alpha.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.726 NppStatus nppiSqrt_32f_AC4R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel image square root with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.727 NppStatus nppiSqrt_32f_C1IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel in place image square root.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.728 NppStatus nppiSqrt_32f_C1R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit floating point channel image square root.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.729 NppStatus nppiSqrt_32f_C3IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel in place image square root.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.730 NppStatus nppiSqrt_32f_C3R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel image square root.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.731 NppStatus nppiSqrt_32f_C4IR (Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel in place image square root.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.732 NppStatus nppiSqrt_32f_C4R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel image square root.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.733 NppStatus nppiSqrt_8u_AC4IRSfs (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel in place image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.734 **NppStatus nppiSqrt_8u_AC4RSfs** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel image square root with unmodified alpha, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.735 **NppStatus nppiSqrt_8u_C1IRSfs** (Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 8-bit unsigned char channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.736 **NppStatus nppiSqrt_8u_C1RSfs** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 8-bit unsigned char channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.737 `NppStatus nppiSqrt_8u_C3IRSfs (Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 8-bit unsigned char channel in place image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.738 `NppStatus nppiSqrt_8u_C3RSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 8-bit unsigned char channel image square root, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.739 `NppStatus nppiSub_16s_AC4IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel with unmodified alpha in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.740 `NppStatus nppiSub_16s_AC4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel with unmodified alpha image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.741 `NppStatus nppiSub_16s_C1IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.742 `NppStatus nppiSub_16s_C1RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.743 `NppStatus nppiSub_16s_C3IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.744 `NppStatus nppiSub_16s_C3RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.745 `NppStatus nppiSub_16s_C4IRSfs (const Npp16s * pSrc, int nSrcStep, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.746 `NppStatus nppiSub_16s_C4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pSrc2, int nSrc2Step, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit signed short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.747 NppStatus nppiSub_16sc_AC4IRSfs (const Npp16sc * pSrc, int nSrcStep, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.748 NppStatus nppiSub_16sc_AC4RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pSrc2, int nSrc2Step, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.749 **NppStatus nppiSub_16sc_C1IRSfs** (const Npp16sc * *pSrc*, int *nSrcStep*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image subtraction, scale by $2^{-(nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.750 **NppStatus nppiSub_16sc_C1RSfs** (const Npp16sc * *pSrc1*, int *nSrc1Step*, const Npp16sc * *pSrc2*, int *nSrc2Step*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image subtraction, scale by $2^{-(nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.751 **NppStatus nppiSub_16sc_C3IRSfs** (const Npp16sc * *pSrc*, int *nSrcStep*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image subtraction, scale by $2^{-(nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.752 `NppStatus nppiSub_16sc_C3RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pSrc2, int nSrc2Step, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.753 `NppStatus nppiSub_16u_AC4IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel with unmodified alpha in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.754 `NppStatus nppiSub_16u_AC4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel with unmodified alpha image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.755 `NppStatus nppiSub_16u_C1IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit unsigned short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.756 `NppStatus nppiSub_16u_C1RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit unsigned short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.757 `NppStatus nppiSub_16u_C3IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit unsigned short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.758 `NppStatus nppiSub_16u_C3RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit unsigned short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.759 `NppStatus nppiSub_16u_C4IRSfs (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.760 `NppStatus nppiSub_16u_C4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.761 NppStatus nppiSub_32f_AC4IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

Four 32-bit floating point channel with unmodified alpha in place image subtraction.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.762 NppStatus nppiSub_32f_AC4R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)

Four 32-bit floating point channel with unmodified alpha image subtraction.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.763 NppStatus nppiSub_32f_C1IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)

One 32-bit floating point channel in place image subtraction.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.764 `NppStatus nppiSub_32f_C1R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel image subtraction.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.765 `NppStatus nppiSub_32f_C3IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel in place image subtraction.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.766 `NppStatus nppiSub_32f_C3R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit floating point channel image subtraction.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.767 `NppStatus nppiSub_32f_C4IR (const Npp32f * pSrc, int nSrcStep, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel in place image subtraction.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.768 `NppStatus nppiSub_32f_C4R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pSrc2, int nSrc2Step, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel image subtraction.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.769 `NppStatus nppiSub_32fc_AC4IR (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image subtraction.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.770 `NppStatus nppiSub_32fc_AC4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pSrc2, int nSrc2Step, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image subtraction.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.771 `NppStatus nppiSub_32fc_C1IR (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image subtraction.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.772 `NppStatus npplSub_32fc_C1R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pSrc2, int nSrc2Step, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image subtraction.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.773 `NppStatus npplSub_32fc_C3IR (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image subtraction.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.774 `NppStatus nppiSub_32fc_C3R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pSrc2, int nSrc2Step, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image subtraction.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.775 `NppStatus nppiSub_32fc_C4IR (const Npp32fc * pSrc, int nSrcStep, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel in place image subtraction.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.776 `NppStatus nppiSub_32fc_C4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pSrc2, int nSrc2Step, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point complex number (32-bit real, 32-bit imaginary) channel image subtraction.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.777 `NppStatus nppiSub_32s_C1IRSfs (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.778 `NppStatus nppiSub_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Note: This function is to be deprecated in future NPP releases, use the function above with a scale factor of 0 instead.

32-bit image subtraction. Subtract pSrc1's pixels from corresponding pixels in pSrc2.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.779 `NppStatus nppiSub_32s_C1RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.780 `NppStatus nppiSub_32s_C3IRSfs (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.781 `NppStatus nppiSub_32s_C3RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.782 `NppStatus nppiSub_32sc_AC4IRSfs (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.783 `NppStatus nppiSub_32sc_AC4RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pSrc2, int nSrc2Step, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel with unmodified alpha image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.784 `NppStatus nppiSub_32sc_C1RSfs (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.785 `NppStatus nppiSub_32sc_C1RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pSrc2, int nSrc2Step, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.786 `NppStatus nppiSub_32sc_C3IRSfs (const Npp32sc * pSrc, int nSrcStep, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.787 `NppStatus nppiSub_32sc_C3RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pSrc2, int nSrc2Step, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer complex number (32-bit real, 32-bit imaginary) channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.788 `NppStatus nppiSub_8u_AC4IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.789 `NppStatus nppiSub_8u_AC4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.790 `NppStatus nppiSub_8u_C1IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.791 `NppStatus nppiSub_8u_C1RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.792 `NppStatus nppiSub_8u_C3IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.793 `NppStatus nppiSub_8u_C3RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 8-bit unsigned char channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.794 `NppStatus nppiSub_8u_C4IRSfs (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel in place image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.795 `NppStatus nppiSub_8u_C4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel image subtraction, scale by $2^{(-nScaleFactor)}$, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.796 NppStatus nppiSubC_16s_AC4IRSfs (const Npp16s * *pConstants*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel with unmodified alpha in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.797 NppStatus nppiSubC_16s_AC4RSfs (const Npp16s * *pSrc1*, int *nSrc1Step*, const Npp16s * *pConstants*, Npp16s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short channel with unmodified alpha image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.798 NppStatus nppiSubC_16s_C1IRSfs (const Npp16s *nConstant*, Npp16s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.799 `NppStatus nppiSubC_16s_C1RSfs (const Npp16s *pSrc1, int nSrc1Step, const Npp16s nConstant, Npp16s *pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.800 `NppStatus nppiSubC_16s_C3IRSfs (const Npp16s *pConstants, Npp16s *pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.801 `NppStatus nppiSubC_16s_C3RSfs (const Npp16s *pSrc1, int nSrc1Step, const Npp16s *pConstants, Npp16s *pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.802 NppStatus nppiSubC_16s_C4IRSfs (const Npp16s * pConstants, Npp16s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit signed short channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.803 NppStatus nppiSubC_16s_C4RSfs (const Npp16s * pSrc1, int nSrc1Step, const Npp16s * pConstants, Npp16s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

Four 16-bit signed short channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.804 **NppStatus nppiSubC_16sc_AC4IRSfs** (const Npp16sc * *pConstants*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.805 **NppStatus nppiSubC_16sc_AC4RSfs** (const Npp16sc * *pSrc1*, int *nSrc1Step*, const Npp16sc * *pConstants*, Npp16sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel with unmodified alpha image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.806 **NppStatus nppiSubC_16sc_C1IRSfs** (const Npp16sc *nConstant*, Npp16sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.807 `NppStatus nppiSubC_16sc_C1RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc nConstant, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.808 `NppStatus nppiSubC_16sc_C3IRSfs (const Npp16sc * pConstants, Npp16sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.809 `NppStatus nppiSubC_16sc_C3RSfs (const Npp16sc * pSrc1, int nSrc1Step, const Npp16sc * pConstants, Npp16sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit signed short complex number (16-bit real, 16-bit imaginary) channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.810 `NppStatus nppiSubC_16u_AC4IRSfs (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel with unmodified alpha in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.811 `NppStatus nppiSubC_16u_AC4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel with unmodified alpha image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.812 NppStatus nppiSubC_16u_C1IRSfs (const Npp16u nConstant, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.813 NppStatus nppiSubC_16u_C1RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u nConstant, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 16-bit unsigned short channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.814 `NppStatus nppiSubC_16u_C3IRSfs (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit unsigned short channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.815 `NppStatus nppiSubC_16u_C3RSfs (const Npp16u * pSrcI, int nSrcIStep, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 16-bit unsigned short channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrcI [Source-Image Pointer](#).
nSrcIStep [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.816 `NppStatus nppiSubC_16u_C4IRSfs (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.817 `NppStatus nppiSubC_16u_C4RSfs (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 16-bit unsigned short channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.818 `NppStatus nppiSubC_32f_AC4IR (const Npp32f * pConstants, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel with unmodified alpha in place image subtract constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.819 `NppStatus nppiSubC_32f_AC4R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f * pConstants, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit floating point channel with unmodified alpha image subtract constant.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.820 `NppStatus nppiSubC_32f_C1IR (const Npp32f nConstant, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel in place image subtract constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.821 `NppStatus nppiSubC_32f_C1R (const Npp32f * pSrc1, int nSrc1Step, const Npp32f nConstant, Npp32f * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit floating point channel image subtract constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.822 `NppStatus nppiSubC_32f_C3IR (const Npp32f * pConstants, Npp32f * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 32-bit floating point channel in place image subtract constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.823 NppStatus nppiSubC_32f_C3R (const Npp32f * *pSrc1*, int *nSrc1Step*, const Npp32f * *pConstants*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit floating point channel image subtract constant.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.824 NppStatus nppiSubC_32f_C4IR (const Npp32f * *pConstants*, Npp32f * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel in place image subtract constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.825 NppStatus nppiSubC_32f_C4R (const Npp32f * *pSrc1*, int *nSrc1Step*, const Npp32f * *pConstants*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit floating point channel image subtract constant.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.826 `NppStatus nppiSubC_32fc_AC4IR (const Npp32fc * pConstants, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha in place image subtract constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.827 `NppStatus nppiSubC_32fc_AC4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pConstants, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel with unmodified alpha image subtract constant.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.828 `NppStatus nppiSubC_32fc_C1IR (const Npp32fc nConstant, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image subtract constant.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.829 NppStatus nppiSubC_32fc_C1R (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc *nConstant*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

One 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image subtract constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.830 NppStatus nppiSubC_32fc_C3IR (const Npp32fc * *pConstants*, Npp32fc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image subtract constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.831 NppStatus nppiSubC_32fc_C3R (const Npp32fc * *pSrc1*, int *nSrc1Step*, const Npp32fc * *pConstants*, Npp32fc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image subtract constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.832 `NppStatus nppiSubC_32fc_C4IR (const Npp32fc * pConstants, Npp32fc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel in place image subtract constant.

Parameters:

pConstants [pointer to a list of constant values, one per channel.](#)

pSrcDst [In-Place Image Pointer.](#)

nSrcDstStep [In-Place-Image Line Step.](#)

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.833 `NppStatus nppiSubC_32fc_C4R (const Npp32fc * pSrc1, int nSrc1Step, const Npp32fc * pConstants, Npp32fc * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit complex floating point (32-bit floating point real, 32-bit floating point imaginary) channel image subtract constant.

Parameters:

pSrc1 [Source-Image Pointer.](#)

nSrc1Step [Source-Image Line Step.](#)

pConstants [pointer to a list of constant values, one per channel.](#)

pDst [Destination-Image Pointer.](#)

nDstStep [Destination-Image Line Step.](#)

oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.834 `NppStatus nppiSubC_32s_C1IRSfs (const Npp32s nConstant, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

nConstant [Constant.](#)

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.835 `NppStatus nppiSubC_32s_C1RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s nConstant, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 32-bit signed integer channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.836 `NppStatus nppiSubC_32s_C3IRSfs (const Npp32s * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.837 `NppStatus nppiSubC_32s_C3RSfs (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 32-bit signed integer channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.838 `NppStatus nppiSubC_32sc_AC4IRSfs (const Npp32sc * pConstants, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst [In-Place Image Pointer](#).
nSrcDstStep [In-Place-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.839 `NppStatus nppiSubC_32sc_AC4RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc * pConstants, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel with unmodified alpha image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).
nSrc1Step [Source-Image Line Step](#).
pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.840 NppStatus nppiSubC_32sc_C1RSfs (const Npp32sc nConstant, Npp32sc * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)

One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.841 NppStatus nppiSubC_32sc_C1RSfs (const Npp32sc * pSrc1, int nSrc1Step, const Npp32sc nConstant, Npp32sc * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)

One 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

nConstant Constant.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.842 **NppStatus nppiSubC_32sc_C3IRSfs** (const Npp32sc * *pConstants*, Npp32sc * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep [In-Place-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.843 **NppStatus nppiSubC_32sc_C3RSfs** (const Npp32sc * *pSrc1*, int *nSrc1Step*, const Npp32sc * *pConstants*, Npp32sc * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Three 32-bit signed complex integer (32-bit real, 32-bit imaginary) channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrc1Step [Source-Image Line Step](#).

pConstants pointer to a list of constant values, one per channel.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.844 **NppStatus nppiSubC_8u_AC4IRSfs** (const Npp8u * *pConstants*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel with unmodified alpha in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst [In-Place Image Pointer](#).

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.845 `NppStatus nppiSubC_8u_AC4RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Four 8-bit unsigned char channel with unmodified alpha image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.846 `NppStatus nppiSubC_8u_C1IRSfs (const Npp8u nConstant, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

nConstant Constant.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.847 `NppStatus nppiSubC_8u_C1RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * nConstant, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

One 8-bit unsigned char channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.848 `NppStatus nppiSubC_8u_C3IRSfs (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 8-bit unsigned char channel 8-bit unsigned char in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).
nScaleFactor Integer Result Scaling.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.849 `NppStatus nppiSubC_8u_C3RSfs (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, int nScaleFactor)`

Three 8-bit unsigned char channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.850 NppStatus nppiSubC_8u_C4IRSfs (const Npp8u * *pConstants*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel in place image subtract constant, scale, then clamp to saturated value.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.851 NppStatus nppiSubC_8u_C4RSfs (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pConstants*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, int *nScaleFactor*)

Four 8-bit unsigned char channel image subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

nScaleFactor Integer Result Scaling.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.852 **NppStatus nppiXor_16u_AC4IR** (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel in place image logical exclusive or with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.853 **NppStatus nppiXor_16u_AC4R** (const Npp16u * *pSrc1*, int *nSrc1Step*, const Npp16u * *pSrc2*, int *nSrc2Step*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 16-bit unsigned short channel image logical exclusive or with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.854 **NppStatus nppiXor_16u_C1IR** (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 16-bit unsigned short channel in place image logical exclusive or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.855 `NppStatus nppiXor_16u_C1R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel image logical exclusive or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.856 `NppStatus nppiXor_16u_C3IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel in place image logical exclusive or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.857 `NppStatus nppiXor_16u_C3R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel image logical exclusive or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.858 `NppStatus nppiXor_16u_C4IR (const Npp16u * pSrc, int nSrcStep, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel in place image logical exclusive or.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.859 `NppStatus nppiXor_16u_C4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pSrc2, int nSrc2Step, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image logical exclusive or.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.860 NppStatus nppiXor_32s_AC4IR (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel in place image logical exclusive or with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.861 NppStatus nppiXor_32s_AC4R (const Npp32s * *pSrc1*, int *nSrc1Step*, const Npp32s * *pSrc2*, int *nSrc2Step*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel image logical exclusive or with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.862 NppStatus nppiXor_32s_C1IR (const Npp32s * *pSrc*, int *nSrcStep*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 32-bit signed integer channel in place image logical exclusive or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.863 `NppStatus nppiXor_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit signed integer channel image logical exclusive or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.864 `NppStatus nppiXor_32s_C3IR (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel in place image logical exclusive or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.865 `NppStatus nppiXor_32s_C3R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Three 32-bit signed integer channel image logical exclusive or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.866 `NppStatus nppiXor_32s_C4IR (const Npp32s * pSrc, int nSrcStep, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel in place image logical exclusive or.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.867 `NppStatus nppiXor_32s_C4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pSrc2, int nSrc2Step, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image logical exclusive or.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.868 NppStatus nppiXor_8u_AC4IR (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel in place image logical exclusive or with unmodified alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.869 NppStatus nppiXor_8u_AC4R (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pSrc2*, int *nSrc2Step*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Four 8-bit unsigned char channel image logical exclusive or with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.870 NppStatus nppiXor_8u_C1IR (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

One 8-bit unsigned char channel in place image logical exclusive or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.871 `NppStatus nppiXor_8u_C1R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel image logical exclusive or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.872 `NppStatus nppiXor_8u_C3IR (const Npp8u * pSrc, int nSrcStep, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel in place image logical exclusive or.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.7.1.873 `NppStatus nppiXor_8u_C3R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pSrc2, int nSrc2Step, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel image logical exclusive or.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.874 `NppStatus nppiXor_8u_C4IR (const Npp8u *pSrc, int nSrcStep, Npp8u *pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image logical exclusive or.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.875 `NppStatus nppiXor_8u_C4R (const Npp8u *pSrc1, int nSrc1Step, const Npp8u *pSrc2, int nSrc2Step, Npp8u *pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image logical exclusive or.

Parameters:

pSrc1 Source-Image Pointer.

nSrc1Step Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrc2Step Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.876 `NppStatus nppiXorC_16u_AC4IR (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel in place image logical exclusive or with constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.877 `NppStatus nppiXorC_16u_AC4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image logical exclusive or with constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.878 `NppStatus nppiXorC_16u_C1IR (const Npp16u nConstant, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel in place image logical exclusive or with constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.879 `NppStatus nppiXorC_16u_C1R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u nConstant, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

One 16-bit unsigned short channel image logical exclusive or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.880 `NppStatus nppiXorC_16u_C3IR (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel in place image logical exclusive or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.881 `NppStatus nppiXorC_16u_C3R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 16-bit unsigned short channel image logical exclusive or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.882 `NppStatus nppiXorC_16u_C4IR (const Npp16u * pConstants, Npp16u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel in place image logical exclusive or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.883 `NppStatus nppiXorC_16u_C4R (const Npp16u * pSrc1, int nSrc1Step, const Npp16u * pConstants, Npp16u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 16-bit unsigned short channel image logical exclusive or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.884 `NppStatus nppiXorC_32s_AC4IR (const Npp32s * pConstants, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel in place image logical exclusive or with constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.885 `NppStatus nppiXorC_32s_AC4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image logical exclusive or with constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.886 `NppStatus nppiXorC_32s_C1IR (const Npp32s nConstant, Npp32s * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 32-bit signed integer channel in place image logical exclusive or with constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.887 `NppStatus nppiXorC_32s_C1R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s nConstant, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

One 32-bit signed integer channel image logical exclusive or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.888 NppStatus nppiXorC_32s_C3IR (const Npp32s * *pConstants*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Three 32-bit signed integer channel in place image logical exclusive or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.889 NppStatus nppiXorC_32s_C3R (const Npp32s * *pSrcI*, int *nSrcIStep*, const Npp32s * *pConstants*, Npp32s * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

Three 32-bit signed integer channel image logical exclusive or with constant.

Parameters:

pSrcI Source-Image Pointer.

nSrcIStep Source-Image Line Step.

pConstants pointer to a list of constant values, one per channel.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.890 NppStatus nppiXorC_32s_C4IR (const Npp32s * *pConstants*, Npp32s * *pSrcDst*, int *nSrcDstStep*, NppiSize *oSizeROI*)

Four 32-bit signed integer channel in place image logical exclusive or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.

pSrcDst In-Place Image Pointer.

nSrcDstStep In-Place-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.891 `NppStatus nppiXorC_32s_C4R (const Npp32s * pSrc1, int nSrc1Step, const Npp32s * pConstants, Npp32s * pDst, int nDstStep, NppiSize oSizeROI)`

Four 32-bit signed integer channel image logical exclusive or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.892 `NppStatus nppiXorC_8u_AC4IR (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image logical exclusive or with constant with unmodified alpha.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.893 `NppStatus nppiXorC_8u_AC4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image logical exclusive or with constant with unmodified alpha.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.894 `NppStatus nppiXorC_8u_C1IR (const Npp8u nConstant, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel in place image logical exclusive or with constant.

Parameters:

nConstant Constant.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.895 `NppStatus nppiXorC_8u_C1R (const Npp8u * pSrcI, int nSrcIStep, const Npp8u nConstant, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

One 8-bit unsigned char channel image logical exclusive or with constant.

Parameters:

pSrcI Source-Image Pointer.
nSrcIStep Source-Image Line Step.
nConstant Constant.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.896 `NppStatus nppiXorC_8u_C3IR (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel in place image logical exclusive or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.897 `NppStatus nppiXorC_8u_C3R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Three 8-bit unsigned char channel image logical exclusive or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.898 `NppStatus nppiXorC_8u_C4IR (const Npp8u * pConstants, Npp8u * pSrcDst, int nSrcDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel in place image logical exclusive or with constant.

Parameters:

pConstants pointer to a list of constant values, one per channel.
pSrcDst In-Place Image Pointer.
nSrcDstStep In-Place-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.7.1.899 `NppStatus nppiXorC_8u_C4R (const Npp8u * pSrc1, int nSrc1Step, const Npp8u * pConstants, Npp8u * pDst, int nDstStep, NppiSize oSizeROI)`

Four 8-bit unsigned char channel image logical exclusive or with constant.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pConstants pointer to a list of constant values, one per channel.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8 Threshold and Compare Operations

Methods for pixel-wise threshold and compare operations.

Threshold

Threshold pixels.

- **NppStatus nppiThreshold_32f_C1R** (const **Npp32f** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI, **Npp32f** nThreshold, **NppCmpOp** eComparisonOperation)
32-bit floating point threshold.
- **NppStatus nppiThreshold_8u_AC4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, const **Npp8u** aThresholds[3], **NppCmpOp** eComparisonOperation)
4 channel 8-bit unsigned image threshold, not affecting Alpha.

Image Compare Methods

Compare the pixels of two images and create a binary result image.

In case of multi-channel image types, the condition must be fulfilled for all channels, otherwise the comparison is considered false. The "binary" result image is of type 8u_C1. False is represented by 0, true by NPP_MAX_8U.

- **NppStatus nppiCompare_8u_C4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppCmpOp** eComparisonOperation)
4 channel 8-bit unsigned image compare.
- **NppStatus nppiCompare_8u_AC4R** (const **Npp8u** *pSrc1, int nSrc1Step, const **Npp8u** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppCmpOp** eComparisonOperation)
4 channel 8-bit unsigned image compare, not affecting Alpha.
- **NppStatus nppiCompare_32f_C1R** (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppCmpOp** eComparisonOperation)
32-bit floating point image compare.

7.8.1 Detailed Description

Methods for pixel-wise threshold and compare operations.

7.8.2 Function Documentation

7.8.2.1 NppStatus nppiCompare_32f_C1R (const **Npp32f** *pSrc1, int nSrc1Step, const **Npp32f** *pSrc2, int nSrc2Step, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppCmpOp** eComparisonOperation)

32-bit floating point image compare.

Compare pSrc1's pixels with corresponding pixels in pSrc2.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eComparisonOperation Specifies the comparison operation to be used in the pixel comparison.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.2.2 NppStatus nppiCompare_8u_AC4R (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pSrc2*, int *nSrc2Step*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, NppCmpOp *eComparisonOperation*)

4 channel 8-bit unsigned image compare, not affecting Alpha.

Compare *pSrc1*'s pixels with corresponding pixels in *pSrc2*.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.
pSrc2 Source-Image Pointer.
nSrc2Step Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).
eComparisonOperation Specifies the comparison operation to be used in the pixel comparison.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.8.2.3 NppStatus nppiCompare_8u_C4R (const Npp8u * *pSrc1*, int *nSrc1Step*, const Npp8u * *pSrc2*, int *nSrc2Step*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, NppCmpOp *eComparisonOperation*)

4 channel 8-bit unsigned image compare.

Compare *pSrc1*'s pixels with corresponding pixels in *pSrc2*.

Parameters:

pSrc1 Source-Image Pointer.
nSrc1Step Source-Image Line Step.

pSrc2 [Source-Image Pointer](#).

nSrc2Step [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

eComparisonOperation Specifies the comparison operation to be used in the pixel comparison.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.8.2.4 NppStatus npptThreshold_32f_C1R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, Npp32f nThreshold, NppCmpOp eComparisonOperation)

32-bit floating point threshold.

If for a comparison operations OP the predicate (sourcePixel OP nThreshold) is true, the pixel is set to nThreshold, otherwise it is set to sourcePixel.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

nThreshold The threshold value.

eComparisonOperation The type of comparison operation to be used. The only valid values are: NPP_CMP_LESS and NPP_CMP_GREATER.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), or NPP_NOT_SUPPORTED_MODE_ERROR if an invalid comparison operation type is specified.

7.8.2.5 NppStatus npptThreshold_8u_AC4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u aThresholds[3], NppCmpOp eComparisonOperation)

4 channel 8-bit unsigned image threshold, not affecting Alpha.

If for a comparison operations OP the predicate (sourcePixel.channel OP nThreshold) is true, the channel value is set to nThreshold, otherwise it is set to sourcePixel.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aThresholds The threshold values, one per color channel.

eComparisonOperation The type of comparison operation to be used. The only valid values are: NPP_CMP_LESS and NPP_CMP_GREATER.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), or NPP_NOT_SUPPORTED_MODE_ERROR if an invalid comparison operation type is specified.

7.9 Statistics Functions

Routines computing statistical image information.

Functions

- **NppStatus nppiSqrIntegral_8u32s32f_C1R** (**Npp8u** *pSrc, int nSrcStep, **Npp32s** *pDst, int nDstStep, **Npp32f** *pSqr, int nSqrStep, **NppiSize** oSrcROI, **Npp32s** val, **Npp32f** valSqr, **Npp32s** integral-ImageNewHeight)

SqrIntegral Transforms an image to integral and integral of pixel squares representation.

- **NppStatus nppiRectStdDev_32s32f_C1R** (const **Npp32s** *pSrc, int nSrcStep, const **Npp64f** *pSqr, int nSqrStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI, **NppiRect** rect)

RectStdDev Computes the standard deviation of integral images.

Mean_StdDev

Computes the mean and standard deviation of image pixel values

- **NppStatus nppiMeanStdDev8uC1RGetBufferHostSize** (**NppiSize** oSizeROI, int *hpBufferSize)
- **NppStatus nppiMean_StdDev_8u_C1R** (const **Npp8u** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp8u** *pDeviceBuffer, **Npp64f** *pMean, **Npp64f** *pStdDev)

8-bit unsigned mean standard deviation.

NormDiff

Norm of pixel differences between two images.

- **NppStatus nppiNormDiff_L1_8u_C1R** (const **Npp8u** *pSrc1, int nSrcStep1, const **Npp8u** *pSrc2, int nSrcStep2, **NppiSize** oSizeROI, **Npp64f** *pRetVal)
- **NppStatus nppiNormDiff_L2_8u_C1R** (const **Npp8u** *pSrc1, int nSrcStep1, const **Npp8u** *pSrc2, int nSrcStep2, **NppiSize** oSizeROI, **Npp64f** *pRetVal)
- **NppStatus nppiNormDiff_Inf_8u_C1R** (const **Npp8u** *pSrc1, int nSrcStep1, const **Npp8u** *pSrc2, int nSrcStep2, **NppiSize** oSizeROI, **Npp64f** *pRetVal)

8-bit unsigned L1 norm of pixel differences.

8-bit unsigned L2 norm of pixel differences.

8-bit unsigned Infinity Norm of pixel differences.

Sum

Sum of 8 bit images.

- [NppStatus nppiReductionGetBufferHostSize_8u_C1R](#) ([NppiSize](#) oSizeROI, int *hpBufferSize)
Scratch-buffer size for nppiSum_8u_C1R.
- [NppStatus nppiReductionGetBufferHostSize_8u_C4R](#) ([NppiSize](#) oSizeROI, int *hpBufferSize)
Scratch-buffer size for nppiSum_8u_C4R.
- [NppStatus nppiSum_8u64s_C1R](#) (const [Npp8u](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp8u](#) *pDeviceBuffer, [Npp64s](#) *pSum)
8-bit unsigned image sum with 64-bit long long result.
- [NppStatus nppiSum_8u_C1R](#) (const [Npp8u](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp8u](#) *pDeviceBuffer, [Npp64f](#) *pSum)
8-bit unsigned image sum with 64-bit double precision result.
- [NppStatus nppiSum_8u64s_C4R](#) (const [Npp8u](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp8u](#) *pDeviceBuffer, [Npp64s](#) aSum[4])
4 channel 8-bit unsigned image sum with 64-bit long long result.
- [NppStatus nppiSum_8u_C4R](#) (const [Npp8u](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp8u](#) *pDeviceBuffer, [Npp64f](#) aSum[4])
4 channel 8-bit unsigned image sum with 64-bit double precision result.

MinMax

Minimum and maximum of 8-bit images.

- [NppStatus nppiMinMaxGetBufferSize_8u_C1R](#) ([NppiSize](#) oSizeROI, int *hpBufferSize)
Scratch-buffer size for nppiMinManx_8u_C1R.
- [NppStatus nppiMinMax_8u_C1R](#) (const [Npp8u](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp8u](#) *pMin, [Npp8u](#) *pMax, [Npp8u](#) *pDeviceBuffer)
8-bit unsigned pixel minimum and maximum.
- [NppStatus nppiMinMaxGetBufferSize_8u_C4R](#) ([NppiSize](#) oSizeROI, int *hpBufferSize)
Scratch-buffer size for nppiMinManx_8u_C4R.
- [NppStatus nppiMinMax_8u_C4R](#) (const [Npp8u](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp8u](#) aMin[4], [Npp8u](#) aMax[4], [Npp8u](#) *pDeviceBuffer)
4 channel 8-bit unsigned pixel minimum and maximum.

Histogram

- [NppStatus nppiEvenLevelsHost_32s](#) ([Npp32s](#) *hpLevels, int nLevels, [Npp32s](#) nLowerLevel, [Npp32s](#) nUpperLevel)
Compute levels with even distribution.
- [NppStatus nppiHistogramEvenGetBufferSize_8u_C1R](#) ([NppiSize](#) oSizeROI, int nLevels, int *hpBufferSize)

Scratch-buffer size for nppiHistogramEven_8u_C1R.

- **NppStatus** **nppiHistogramEven_8u_C1R** (const **Npp8u** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist, int nLevels, **Npp32s** nLowerLevel, **Npp32s** nUpperLevel, **Npp8u** *pBuffer)

8-bit unsigned histogram with evenly distributed bins.

- **NppStatus** **nppiHistogramEvenGetBufferSize_8u_C4R** (**NppiSize** oSizeROI, int nLevels[4], int *hpBufferSize)

Scratch-buffer size for nppiHistogramEven_8u_C4R.

- **NppStatus** **nppiHistogramEven_8u_C4R** (const **Npp8u** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist[4], int nLevels[4], **Npp32s** nLowerLevel[4], **Npp32s** nUpperLevel[4], **Npp8u** *pBuffer)

4 channel 8-bit unsigned histogram with evenly distributed bins.

- **NppStatus** **nppiHistogramEvenGetBufferSize_8u_AC4R** (**NppiSize** oSizeROI, int nLevels[3], int *hpBufferSize)

Scratch-buffer size for nppiHistogramEven_8u_AC4R.

- **NppStatus** **nppiHistogramEven_8u_AC4R** (const **Npp8u** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist[3], int nLevels[3], **Npp32s** nLowerLevel[3], **Npp32s** nUpperLevel[3], **Npp8u** *pBuffer)

4 channel (alpha as the last channel) 8-bit unsigned histogram with evenly distributed bins.

- **NppStatus** **nppiHistogramEvenGetBufferSize_16u_C1R** (**NppiSize** oSizeROI, int nLevels, int *hpBufferSize)

Scratch-buffer size for nppiHistogramEven_16u_C1R.

- **NppStatus** **nppiHistogramEven_16u_C1R** (const **Npp16u** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist, int nLevels, **Npp32s** nLowerLevel, **Npp32s** nUpperLevel, **Npp8u** *pBuffer)

16-bit unsigned histogram with evenly distributed bins.

- **NppStatus** **nppiHistogramEvenGetBufferSize_16u_C4R** (**NppiSize** oSizeROI, int nLevels[4], int *hpBufferSize)

Scratch-buffer size for nppiHistogramEven_16u_C4R.

- **NppStatus** **nppiHistogramEven_16u_C4R** (const **Npp16u** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist[4], int nLevels[4], **Npp32s** nLowerLevel[4], **Npp32s** nUpperLevel[4], **Npp8u** *pBuffer)

4 channel 16-bit unsigned histogram with evenly distributed bins.

- **NppStatus** **nppiHistogramEvenGetBufferSize_16u_AC4R** (**NppiSize** oSizeROI, int nLevels[3], int *hpBufferSize)

Scratch-buffer size for nppiHistogramEven_16u_AC4R.

- **NppStatus** **nppiHistogramEven_16u_AC4R** (const **Npp16u** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist[3], int nLevels[3], **Npp32s** nLowerLevel[3], **Npp32s** nUpperLevel[3], **Npp8u** *pBuffer)

4 channel (alpha as the last channel) 16-bit unsigned histogram with evenly distributed bins.

- [NppStatus nppiHistogramEvenGetBufferSize_16s_C1R](#) ([NppiSize](#) oSizeROI, int nLevels, int *hpBufferSize)
Scratch-buffer size for nppiHistogramEven_16s_C1R.
- [NppStatus nppiHistogramEven_16s_C1R](#) (const [Npp16s](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp32s](#) *pHist, int nLevels, [Npp32s](#) nLowerLevel, [Npp32s](#) nUpperLevel, [Npp8u](#) *pBuffer)
16-bit signed histogram with evenly distributed bins.
- [NppStatus nppiHistogramEvenGetBufferSize_16s_C4R](#) ([NppiSize](#) oSizeROI, int nLevels[4], int *hpBufferSize)
Scratch-buffer size for nppiHistogramEven_16s_C4R.
- [NppStatus nppiHistogramEven_16s_C4R](#) (const [Npp16s](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp32s](#) *pHist[4], int nLevels[4], [Npp32s](#) nLowerLevel[4], [Npp32s](#) nUpperLevel[4], [Npp8u](#) *pBuffer)
4 channel 16-bit signed histogram with evenly distributed bins.
- [NppStatus nppiHistogramEvenGetBufferSize_16s_AC4R](#) ([NppiSize](#) oSizeROI, int nLevels[3], int *hpBufferSize)
Scratch-buffer size for nppiHistogramEven_16s_AC4R.
- [NppStatus nppiHistogramEven_16s_AC4R](#) (const [Npp16s](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp32s](#) *pHist[3], int nLevels[3], [Npp32s](#) nLowerLevel[3], [Npp32s](#) nUpperLevel[3], [Npp8u](#) *pBuffer)
4 channel (alpha as the last channel) 16-bit signed histogram with evenly distributed bins.
- [NppStatus nppiHistogramRangeGetBufferSize_8u_C1R](#) ([NppiSize](#) oSizeROI, int nLevels, int *hpBufferSize)
Scratch-buffer size for nppiHistogramRange_8u_C1R.
- [NppStatus nppiHistogramRange_8u_C1R](#) (const [Npp8u](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp32s](#) *pHist, const [Npp32s](#) *pLevels, int nLevels, [Npp8u](#) *pBuffer)
8-bit unsigned histogram with bins determined by pLevels array.
- [NppStatus nppiHistogramRangeGetBufferSize_8u_C4R](#) ([NppiSize](#) oSizeROI, int nLevels[4], int *hpBufferSize)
Scratch-buffer size for nppiHistogramRange_8u_C4R.
- [NppStatus nppiHistogramRange_8u_C4R](#) (const [Npp8u](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp32s](#) *pHist[4], const [Npp32s](#) *pLevels[4], int nLevels[4], [Npp8u](#) *pBuffer)
4 channel 8-bit unsigned histogram with bins determined by pLevels.
- [NppStatus nppiHistogramRangeGetBufferSize_8u_AC4R](#) ([NppiSize](#) oSizeROI, int nLevels[3], int *hpBufferSize)
Scratch-buffer size for nppiHistogramRange_8u_AC4R.
- [NppStatus nppiHistogramRange_8u_AC4R](#) (const [Npp8u](#) *pSrc, int nSrcStep, [NppiSize](#) oSizeROI, [Npp32s](#) *pHist[3], const [Npp32s](#) *pLevels[3], int nLevels[3], [Npp8u](#) *pBuffer)
4 channel (alpha as a last channel) 8-bit unsigned histogram with bins determined by pLevels.

- **NppStatus nppiHistogramRangeGetBufferSize_16u_C1R** (**NppiSize** oSizeROI, int nLevels, int *hpBufferSize)
Scratch-buffer size for nppiHistogramRange_16u_C1R.
- **NppStatus nppiHistogramRange_16u_C1R** (const **Npp16u** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist, const **Npp32s** *pLevels, int nLevels, **Npp8u** *pBuffer)
16-bit unsigned histogram with bins determined by pLevels array.
- **NppStatus nppiHistogramRangeGetBufferSize_16u_C4R** (**NppiSize** oSizeROI, int nLevels[4], int *hpBufferSize)
Scratch-buffer size for nppiHistogramRange_16u_C4R.
- **NppStatus nppiHistogramRange_16u_C4R** (const **Npp16u** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist[4], const **Npp32s** *pLevels[4], int nLevels[4], **Npp8u** *pBuffer)
4 channel 16-bit unsigned histogram with bins determined by pLevels.
- **NppStatus nppiHistogramRangeGetBufferSize_16u_AC4R** (**NppiSize** oSizeROI, int nLevels[3], int *hpBufferSize)
Scratch-buffer size for nppiHistogramRange_16u_AC4R.
- **NppStatus nppiHistogramRange_16u_AC4R** (const **Npp16u** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist[3], const **Npp32s** *pLevels[3], int nLevels[3], **Npp8u** *pBuffer)
4 channel (alpha as a last channel) 16-bit unsigned histogram with bins determined by pLevels.
- **NppStatus nppiHistogramRangeGetBufferSize_16s_C1R** (**NppiSize** oSizeROI, int nLevels, int *hpBufferSize)
Scratch-buffer size for nppiHistogramRange_16s_C1R.
- **NppStatus nppiHistogramRange_16s_C1R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist, const **Npp32s** *pLevels, int nLevels, **Npp8u** *pBuffer)
16-bit signed histogram with bins determined by pLevels array.
- **NppStatus nppiHistogramRangeGetBufferSize_16s_C4R** (**NppiSize** oSizeROI, int nLevels[4], int *hpBufferSize)
Scratch-buffer size for nppiHistogramRange_16s_C4R.
- **NppStatus nppiHistogramRange_16s_C4R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist[4], const **Npp32s** *pLevels[4], int nLevels[4], **Npp8u** *pBuffer)
4 channel 16-bit signed histogram with bins determined by pLevels.
- **NppStatus nppiHistogramRangeGetBufferSize_16s_AC4R** (**NppiSize** oSizeROI, int nLevels[3], int *hpBufferSize)
Scratch-buffer size for nppiHistogramRange_16s_AC4R.
- **NppStatus nppiHistogramRange_16s_AC4R** (const **Npp16s** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist[3], const **Npp32s** *pLevels[3], int nLevels[3], **Npp8u** *pBuffer)
4 channel (alpha as a last channel) 16-bit signed histogram with bins determined by pLevels.
- **NppStatus nppiHistogramRangeGetBufferSize_32f_C1R** (**NppiSize** oSizeROI, int nLevels, int *hpBufferSize)
Scratch-buffer size for nppiHistogramRange_32f_C1R.

- **NppStatus nppiHistogramRange_32f_C1R** (const **Npp32f** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist, const **Npp32f** *pLevels, int nLevels, **Npp8u** *pBuffer)

32-bit float histogram with bins determined by pLevels array.

- **NppStatus nppiHistogramRangeGetBufferSize_32f_C4R** (**NppiSize** oSizeROI, int nLevels[4], int *hpBufferSize)

Scratch-buffer size for nppiHistogramRange_32f_C4R.

- **NppStatus nppiHistogramRange_32f_C4R** (const **Npp32f** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist[4], const **Npp32f** *pLevels[4], int nLevels[4], **Npp8u** *pBuffer)

4 channel 32-bit float histogram with bins determined by pLevels.

- **NppStatus nppiHistogramRangeGetBufferSize_32f_AC4R** (**NppiSize** oSizeROI, int nLevels[3], int *hpBufferSize)

Scratch-buffer size for nppiHistogramRange_32f_AC4R.

- **NppStatus nppiHistogramRange_32f_AC4R** (const **Npp32f** *pSrc, int nSrcStep, **NppiSize** oSizeROI, **Npp32s** *pHist[3], const **Npp32f** *pLevels[3], int nLevels[3], **Npp8u** *pBuffer)

4 channel (alpha as a last channel) 32-bit float histogram with bins determined by pLevels.

7.9.1 Detailed Description

Routines computing statistical image information.

7.9.2 Function Documentation

7.9.2.1 **NppStatus nppiEvenLevelsHost_32s** (**Npp32s** *hpLevels, int nLevels, **Npp32s** nLowerLevel, **Npp32s** nUpperLevel)

Compute levels with even distribution.

Parameters:

hpLevels A host pointer to array which receives the levels being computed. The array needs to be of size nLevels.

nLevels The number of levels being computed. nLevels must be at least 2, otherwise an NPP_HISTO_NUMBER_OF_LEVELS_ERROR error is returned.

nLowerLevel Lower boundary value of the lowest level.

nUpperLevel Upper boundary value of the greatest level.

Returns:

Error code.

7.9.2.2 NppStatus nppiHistogramEven_16s_AC4R (const Npp16s * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp32s * *pHist*[3], int *nLevels*[3], Npp32s *nLowerLevel*[3], Npp32s *nUpperLevel*[3], Npp8u * *pBuffer*)

4 channel (alpha as the last channel) 16-bit signed histogram with evenly distributed bins.

Alpha channel is ignored during histogram computation.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving computed histograms per color channel. Array pointed by *pHist*[*i*] be of size *nLevels*[*i*]-1.

nLevels Array containing number of levels per color channel.

nLowerLevel Array containing lower-level of lowest bin per color channel.

nUpperLevel Array containing upper-level of highest bin per color channel.

pBuffer Pointer to appropriately sized (`nppiHistogramEvenGetBufferSize_16s_AC4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.3 NppStatus nppiHistogramEven_16s_C1R (const Npp16s * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp32s * *pHist*, int *nLevels*, Npp32s *nLowerLevel*, Npp32s *nUpperLevel*, Npp8u * *pBuffer*)

16-bit signed histogram with evenly distributed bins.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Pointer to array that receives the computed histogram. The array must be of size *nLevels*-1.

nLevels Number of levels.

nLowerLevel Lower boundary of lowest level bin.

nUpperLevel Upper boundary of highest level bin.

pBuffer Pointer to appropriately sized (`nppiHistogramEvenGetBufferSize_16s_C1R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.4 `NppStatus nppiHistogramEven_16s_C4R (const Npp16s * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist[4], int nLevels[4], Npp32s nLowerLevel[4], Npp32s nUpperLevel[4], Npp8u * pBuffer)`

4 channel 16-bit signed histogram with evenly distributed bins.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving computed histograms per color channel. Array pointed by pHist[i] be of size nLevels[i]-1.

nLevels Array containing number of levels per color channel.

nLowerLevel Array containing lower-level of lowest bin per color channel.

nUpperLevel Array containing upper-level of highest bin per color channel.

pBuffer Pointer to appropriately sized (`nppiHistogramEvenGetBufferSize_16s_C4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.5 `NppStatus nppiHistogramEven_16u_AC4R (const Npp16u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist[3], int nLevels[3], Npp32s nLowerLevel[3], Npp32s nUpperLevel[3], Npp8u * pBuffer)`

4 channel (alpha as the last channel) 16-bit unsigned histogram with evenly distributed bins.

Alpha channel is ignored during histogram computation.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving computed histograms per color channel. Array pointed by pHist[i] be of size nLevels[i]-1.

nLevels Array containing number of levels per color channel.

nLowerLevel Array containing lower-level of lowest bin per color channel.

nUpperLevel Array containing upper-level of highest bin per color channel.

pBuffer Pointer to appropriately sized (`nppiHistogramEvenGetBufferSize_16u_AC4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.6 `NppStatus nppiHistogramEven_16u_C1R (const Npp16u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist, int nLevels, Npp32s nLowerLevel, Npp32s nUpperLevel, Npp8u * pBuffer)`

16-bit unsigned histogram with evenly distributed bins.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Pointer to array that receives the computed histogram. The array must be of size *nLevels*-1.

nLevels Number of levels.

nLowerLevel Lower boundary of lowest level bin.

nUpperLevel Upper boundary of highest level bin.

pBuffer Pointer to appropriately sized (`nppiHistogramEvenGetBufferSize_16u_C1R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.7 `NppStatus nppiHistogramEven_16u_C4R (const Npp16u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist[4], int nLevels[4], Npp32s nLowerLevel[4], Npp32s nUpperLevel[4], Npp8u * pBuffer)`

4 channel 16-bit unsigned histogram with evenly distributed bins.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving computed histograms per color channel. Array pointed by *pHist*[*i*] be of size *nLevels*[*i*]-1.

nLevels Array containing number of levels per color channel.

nLowerLevel Array containing lower-level of lowest bin per color channel.

nUpperLevel Array containing upper-level of highest bin per color channel.

pBuffer Pointer to appropriately sized (`nppiHistogramEvenGetBufferSize_16u_C4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.8 `NppStatus nppiHistogramEven_8u_AC4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist[3], int nLevels[3], Npp32s nLowerLevel[3], Npp32s nUpperLevel[3], Npp8u * pBuffer)`

4 channel (alpha as the last channel) 8-bit unsigned histogram with evenly distributed bins.

Alpha channel is ignored during histogram computation.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
pHist Array of pointers which are receiving computed histograms per color channel. Array pointed by pHist[i] be of size nLevels[i]-1.
nLevels Array containing number of levels per color channel.
nLowerLevel Array containing lower-level of lowest bin per color channel.
nUpperLevel Array containing upper-level of highest bin per color channel.
pBuffer Pointer to appropriately sized (nppiHistogramEvenGetBufferSize_8u_AC4R) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.9 `NppStatus nppiHistogramEven_8u_C1R (const Npp8u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist, int nLevels, Npp32s nLowerLevel, Npp32s nUpperLevel, Npp8u * pBuffer)`

8-bit unsigned histogram with evenly distributed bins.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).
pHist Pointer to array that receives the computed histogram. The array must be of size nLevels-1.
nLevels Number of levels.
nLowerLevel Lower boundary of lowest level bin.
nUpperLevel Upper boundary of highest level bin.
pBuffer Pointer to appropriately sized (nppiHistogramEvenGetBufferSize_8u_C1R) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.10 `NppStatus nppiHistogramEven_8u_C4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist[4], int nLevels[4], Npp32s nLowerLevel[4], Npp32s nUpperLevel[4], Npp8u * pBuffer)`

4 channel 8-bit unsigned histogram with evenly distributed bins.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving computed histograms per color channel. Array pointed by *pHist[i]* be of size *nLevels[i]-1*.

nLevels Array containing number of levels per color channel.

nLowerLevel Array containing lower-level of lowest bin per color channel.

nUpperLevel Array containing upper-level of highest bin per color channel.

pBuffer Pointer to appropriately sized (`npPiHistogramEvenGetBufferSize_8u_C4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.11 NppStatus npPiHistogramEvenGetBufferSize_16s_AC4R (NppiSize *oSizeROI*, int *nLevels*[3], int * *hpBufferSize*)

Scratch-buffer size for `npPiHistogramEven_16s_AC4R`.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.12 NppStatus npPiHistogramEvenGetBufferSize_16s_C1R (NppiSize *oSizeROI*, int *nLevels*, int * *hpBufferSize*)

Scratch-buffer size for `npPiHistogramEven_16s_C1R`.

Parameters:

oSizeROI [Region-of-Interest \(ROI\)](#).

nLevels Number of levels in the histogram.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.13 NppStatus npPiHistogramEvenGetBufferSize_16s_C4R (NppiSize *oSizeROI*, int *nLevels*[4], int * *hpBufferSize*)

Scratch-buffer size for `npPiHistogramEven_16s_C4R`.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.14 NppStatus nppiHistogramEvenGetBufferSize_16u_AC4R (NppiSize oSizeROI, int nLevels[3], int * hpBufferSize)

Scratch-buffer size for nppiHistogramEven_16u_AC4R.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.15 NppStatus nppiHistogramEvenGetBufferSize_16u_C1R (NppiSize oSizeROI, int nLevels, int * hpBufferSize)

Scratch-buffer size for nppiHistogramEven_16u_C1R.

Parameters:

oSizeROI [Region-of-Interest \(ROI\)](#).

nLevels Number of levels in the histogram.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.16 NppStatus nppiHistogramEvenGetBufferSize_16u_C4R (NppiSize oSizeROI, int nLevels[4], int * hpBufferSize)

Scratch-buffer size for nppiHistogramEven_16u_C4R.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.17 NppStatus nppiHistogramEvenGetBufferSize_8u_AC4R (NppiSize *oSizeROI*, int *nLevels*[3], int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramEven_8u_AC4R.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.18 NppStatus nppiHistogramEvenGetBufferSize_8u_C1R (NppiSize *oSizeROI*, int *nLevels*, int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramEven_8u_C1R.

Parameters:

oSizeROI [Region-of-Interest \(ROI\)](#).

nLevels Number of levels in the histogram.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.19 NppStatus nppiHistogramEvenGetBufferSize_8u_C4R (NppiSize *oSizeROI*, int *nLevels*[4], int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramEven_8u_C4R.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.20 NppStatus nppiHistogramRange_16s_AC4R (const Npp16s * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp32s * *pHist*[3], const Npp32s * *pLevels*[3], int *nLevels*[3], Npp8u * *pBuffer*)

4 channel (alpha as a last channel) 16-bit signed histogram with bins determined by *pLevels*.

Alpha channel is ignored during the histograms computations.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving the computed histograms per color channel. Array pointed by pHist[i] must be of size nLevels[i]-1.

nLevels Array containing number of levels per color channel.

pLevels Array containing pointers to level-arrays per color channel. Array pointed by pLevel[i] must be of size nLevels[i].

pBuffer Pointer to appropriately sized (nppiHistogramRangeGetBufferSize_16_AC4R) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.21 NppStatus nppiHistogramRange_16s_C1R (const Npp16s * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp32s * *pHist*, const Npp32s * *pLevels*, int *nLevels*, Npp8u * *pBuffer*)

16-bit signed histogram with bins determined by pLevels array.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Pointer to array that receives the computed histogram. The array must be of size nLevels-1.

pLevels Pointer to array containing the level sizes of the bins. The array must be of size nLevels.

nLevels Number of levels in histogram.

pBuffer Pointer to appropriately sized (nppiHistogramRangeGetBufferSize_16_C1R) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.22 NppStatus nppiHistogramRange_16s_C4R (const Npp16s * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp32s * *pHist*[4], const Npp32s * *pLevels*[4], int *nLevels*[4], Npp8u * *pBuffer*)

4 channel 16-bit signed histogram with bins determined by pLevels.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving the computed histograms per color channel. Array pointed by pHist[i] must be of size nLevels[i]-1.

nLevels Array containing number of levels per color channel.

pLevels Array containing pointers to level-arrays per color channel. Array pointed by *pLevel[i]* must be of size *nLevels[i]*.

pBuffer Pointer to appropriately sized (`npplHistogramRangeGetBufferSize_16s_C4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.23 `NppStatus npplHistogramRange_16u_AC4R (const Npp16u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist[3], const Npp32s * pLevels[3], int nLevels[3], Npp8u * pBuffer)`

4 channel (alpha as a last channel) 16-bit unsigned histogram with bins determined by *pLevels*.

Alpha channel is ignored during the histograms computations.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving the computed histograms per color channel. Array pointed by *pHist[i]* must be of size *nLevels[i]-1*.

nLevels Array containing number of levels per color channel.

pLevels Array containing pointers to level-arrays per color channel. Array pointed by *pLevel[i]* must be of size *nLevels[i]*.

pBuffer Pointer to appropriately sized (`npplHistogramRangeGetBufferSize_16u_AC4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.24 `NppStatus npplHistogramRange_16u_C1R (const Npp16u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist, const Npp32s * pLevels, int nLevels, Npp8u * pBuffer)`

16-bit unsigned histogram with bins determined by *pLevels* array.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Pointer to array that receives the computed histogram. The array must be of size *nLevels-1*.

pLevels Pointer to array containing the level sizes of the bins. The array must be of size *nLevels*.

nLevels Number of levels in histogram.

pBuffer Pointer to appropriately sized (`npplHistogramRangeGetBufferSize_16u_C1R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.25 `NppStatus nppiHistogramRange_16u_C4R (const Npp16u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist[4], const Npp32s * pLevels[4], int nLevels[4], Npp8u * pBuffer)`

4 channel 16-bit unsigned histogram with bins determined by pLevels.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving the computed histograms per color channel. Array pointed by pHist[i] must be of size nLevels[i]-1.

nLevels Array containing number of levels per color channel.

pLevels Array containing pointers to level-arrays per color channel. Array pointed by pLevel[i] must be of size nLevels[i].

pBuffer Pointer to appropriately sized (`nppiHistogramRangeGetBufferSize_16u_C4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.26 `NppStatus nppiHistogramRange_32f_AC4R (const Npp32f * pSrc, int nSrcStep, NppiSize oSizeROI, Npp32s * pHist[3], const Npp32f * pLevels[3], int nLevels[3], Npp8u * pBuffer)`

4 channel (alpha as a last channel) 32-bit float histogram with bins determined by pLevels.

Alpha channel is ignored during the histograms computations.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving the computed histograms per color channel. Array pointed by pHist[i] must be of size nLevels[i]-1.

nLevels Array containing number of levels per color channel.

pLevels Array containing pointers to level-arrays per color channel. Array pointed by pLevel[i] must be of size nLevels[i].

pBuffer Pointer to appropriately sized (`nppiHistogramRangeGetBufferSize_32f_AC4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.27 **NppStatus nppiHistogramRange_32f_C1R** (const Npp32f * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp32s * *pHist*, const Npp32f * *pLevels*, int *nLevels*, Npp8u * *pBuffer*)

32-bit float histogram with bins determined by *pLevels* array.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Pointer to array that receives the computed histogram. The array must be of size *nLevels*-1.

pLevels Pointer to array containing the level sizes of the bins. The array must be of size *nLevels*.

nLevels Number of levels in histogram.

pBuffer Pointer to appropriately sized (`nppiHistogramRangeGetBufferSize_32f_C1R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.28 **NppStatus nppiHistogramRange_32f_C4R** (const Npp32f * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp32s * *pHist*[4], const Npp32f * *pLevels*[4], int *nLevels*[4], Npp8u * *pBuffer*)

4 channel 32-bit float histogram with bins determined by *pLevels*.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving the computed histograms per color channel. Array pointed by *pHist*[*i*] must be of size *nLevels*[*i*]-1.

nLevels Array containing number of levels per color channel.

pLevels Array containing pointers to level-arrays per color channel. Array pointed by *pLevel*[*i*] must be of size *nLevels*[*i*].

pBuffer Pointer to appropriately sized (`nppiHistogramRangeGetBufferSize_32f_C4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.29 **NppStatus nppiHistogramRange_8u_AC4R** (const Npp8u * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp32s * *pHist*[3], const Npp32s * *pLevels*[3], int *nLevels*[3], Npp8u * *pBuffer*)

4 channel (alpha as a last channel) 8-bit unsigned histogram with bins determined by *pLevels*.

Alpha channel is ignored during the histograms computations.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving the computed histograms per color channel. Array pointed by pHist[i] must be of size nLevels[i]-1.

nLevels Array containing number of levels per color channel.

pLevels Array containing pointers to level-arrays per color channel. Array pointed by pLevel[i] must be of size nLevels[i].

pBuffer Pointer to appropriately sized (nppiHistogramRangeGetBufferSize_8u_AC4R) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.30 **NppStatus nppiHistogramRange_8u_C1R** (const Npp8u * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp32s * *pHist*, const Npp32s * *pLevels*, int *nLevels*, Npp8u * *pBuffer*)

8-bit unsigned histogram with bins determined by pLevels array.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Pointer to array that receives the computed histogram. The array must be of size nLevels-1.

pLevels Pointer to array containing the level sizes of the bins. The array must be of size nLevels.

nLevels Number of levels in histogram.

pBuffer Pointer to appropriately sized (nppiHistogramRangeGetBufferSize_8u_C1R) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.31 **NppStatus nppiHistogramRange_8u_C4R** (const Npp8u * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp32s * *pHist*[4], const Npp32s * *pLevels*[4], int *nLevels*[4], Npp8u * *pBuffer*)

4 channel 8-bit unsigned histogram with bins determined by pLevels.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pHist Array of pointers which are receiving the computed histograms per color channel. Array pointed by pHist[i] must be of size nLevels[i]-1.

nLevels Array containing number of levels per color channel.

pLevels Array containing pointers to level-arrays per color channel. Array pointed by *pLevel[i]* must be of size *nLevels[i]*.

pBuffer Pointer to appropriately sized (`nppiHistogramRangeGetBufferSize_8u_C4R`) scratch buffer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.32 NppStatus nppiHistogramRangeGetBufferSize_16s_AC4R (NppiSize oSizeROI, int nLevels[3], int * hpBufferSize)

Scratch-buffer size for `nppiHistogramRange_16s_AC4R`.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.33 NppStatus nppiHistogramRangeGetBufferSize_16s_C1R (NppiSize oSizeROI, int nLevels, int * hpBufferSize)

Scratch-buffer size for `nppiHistogramRange_16s_C1R`.

Parameters:

oSizeROI [Region-of-Interest \(ROI\)](#).

nLevels Number of levels in the histogram.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.34 NppStatus nppiHistogramRangeGetBufferSize_16s_C4R (NppiSize oSizeROI, int nLevels[4], int * hpBufferSize)

Scratch-buffer size for `nppiHistogramRange_16s_C4R`.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.35 NppStatus nppiHistogramRangeGetBufferSize_16u_AC4R (NppiSize *oSizeROI*, int *nLevels*[3], int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramRange_16u_AC4R.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.36 NppStatus nppiHistogramRangeGetBufferSize_16u_C1R (NppiSize *oSizeROI*, int *nLevels*, int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramRange_16u_C1R.

Parameters:

oSizeROI [Region-of-Interest \(ROI\)](#).

nLevels Number of levels in the histogram.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.37 NppStatus nppiHistogramRangeGetBufferSize_16u_C4R (NppiSize *oSizeROI*, int *nLevels*[4], int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramRange_16u_C4R.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.38 NppStatus nppiHistogramRangeGetBufferSize_32f_AC4R (NppiSize *oSizeROI*, int *nLevels*[3], int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramRange_32f_AC4R.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.39 NppStatus nppiHistogramRangeGetBufferSize_32f_C1R (NppiSize *oSizeROI*, int *nLevels*, int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramRange_32f_C1R.

Parameters:

oSizeROI [Region-of-Interest \(ROI\)](#).

nLevels Number of levels in the histogram.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.40 NppStatus nppiHistogramRangeGetBufferSize_32f_C4R (NppiSize *oSizeROI*, int *nLevels*[4], int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramRange_32f_C4R.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.41 NppStatus nppiHistogramRangeGetBufferSize_8u_AC4R (NppiSize *oSizeROI*, int *nLevels*[3], int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramRange_8u_AC4R.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.42 NppStatus nppiHistogramRangeGetBufferSize_8u_C1R (NppiSize *oSizeROI*, int *nLevels*, int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramRange_8u_C1R.

Parameters:

oSizeROI [Region-of-Interest \(ROI\)](#).

nLevels Number of levels in the histogram.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.43 NppStatus nppiHistogramRangeGetBufferSize_8u_C4R (NppiSize *oSizeROI*, int *nLevels*[4], int * *hpBufferSize*)

Scratch-buffer size for nppiHistogramRange_8u_C4R.

Parameters:

oSizeROI ROI size.

nLevels Array containing number of levels per color channel.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.44 NppStatus nppiMean_StdDev_8u_C1R (const Npp8u * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp8u * *pDeviceBuffer*, Npp64f * *pMean*, Npp64f * *pStdDev*)

8-bit unsigned mean standard deviation.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pDeviceBuffer Pointer to the required device memory allocation, [Scratch Buffer and Host Pointer](#)

pMean Contains computed mean.

pStdDev Contains computed standard deviation.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.45 NppStatus nppiMeanStdDev8uC1RGetBufferHostSize (NppiSize oSizeROI, int * hpBufferSize)

Device scratch buffer size (in bytes) for mean and standard deviation of image.

This primitive provides the correct buffer size for nppiMean_StdDev_8u_C1R.

Parameters:

oSizeROI [Region-of-Interest \(ROI\)](#).

hpBufferSize Required buffer size. Important: hpBufferSize is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.9.2.46 NppStatus nppiMinMax_8u_C1R (const Npp8u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp8u * pMin, Npp8u * pMax, Npp8u * pDeviceBuffer)

8-bit unsigned pixel minimum and maximum.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMin Device-memory pointer receiving the minimum result.

pMax Device-memory pointer receiving the maximum result.

pDeviceBuffer Buffer to a scratch memory. Use [nppiMinMaxGetBufferSize_8u_C1R](#) to determine the minium number of bytes required.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.47 NppStatus nppiMinMax_8u_C4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp8u aMin[4], Npp8u aMax[4], Npp8u * pDeviceBuffer)

4 channel 8-bit unsigned pixel minimum and maximum.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

aMin Device-pointer (array) receiving the minimum result.

aMax Device-pointer (array) receiving the maximum result.

pDeviceBuffer Buffer to a scratch memory. Use [nppiMinMaxGetBufferSize_8u_C4R](#) to determine the minium number of bytes required.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

Note:

Unlike `nppiMinMax_8u_C1R`, this primitive returns its results as device pointers.

7.9.2.48 NppStatus nppiMinMaxGetBufferSize_8u_C1R (NppiSize oSizeROI, int * hpBufferSize)

Scratch-buffer size for `nppiMinMax_8u_C1R`.

Parameters:

oSizeROI ROI size.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.49 NppStatus nppiMinMaxGetBufferSize_8u_C4R (NppiSize oSizeROI, int * hpBufferSize)

Scratch-buffer size for `nppiMinMax_8u_C4R`.

Parameters:

oSizeROI ROI size.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.50 NppStatus nppiNormDiff_Inf_8u_C1R (const Npp8u * pSrc1, int nSrcStep1, const Npp8u * pSrc2, int nSrcStep2, NppiSize oSizeROI, Npp64f * pRetVal)

8-bit unsigned Infinity Norm of pixel differences.

Parameters:

pSrc1 [Source-Image Pointer](#).

nSrcStep1 [Source-Image Line Step](#).

pSrc2 [Source-Image Pointer](#).

nSrcStep2 [Source-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

**pRetVal* Contains computed L1-norm of differences. This is a host pointer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.51 `NppStatus nppiNormDiff_L1_8u_C1R (const Npp8u * pSrc1, int nSrcStep1, const Npp8u * pSrc2, int nSrcStep2, NppiSize oSizeROI, Npp64f * pRetVal)`

8-bit unsigned L1 norm of pixel differences.

Parameters:

pSrc1 Source-Image Pointer.

nSrcStep1 Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrcStep2 Source-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pRetVal Contains computed L1-norm of differences. This is a host pointer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.52 `NppStatus nppiNormDiff_L2_8u_C1R (const Npp8u * pSrc1, int nSrcStep1, const Npp8u * pSrc2, int nSrcStep2, NppiSize oSizeROI, Npp64f * pRetVal)`

8-bit unsigned L2 norm of pixel differences.

Parameters:

pSrc1 Source-Image Pointer.

nSrcStep1 Source-Image Line Step.

pSrc2 Source-Image Pointer.

nSrcStep2 Source-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pRetVal Contains computed L1-norm of differences. This is a host pointer.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.53 `NppStatus nppiRectStdDev_32s32f_C1R (const Npp32s * pSrc, int nSrcStep, const Npp64f * pSqr, int nSqrStep, Npp32f * pDst, int nDstStep, NppiSize oSizeROI, NppiRect rect)`

RectStdDev Computes the standard deviation of integral images.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pSqr Destination-Image Pointer.

nSqrStep Destination-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

rect rectangular window

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.2.54 NppStatus nppiReductionGetBufferHostSize_8u_C1R (NppiSize oSizeROI, int * hpBufferSize)

Scratch-buffer size for nppiSum_8u_C1R.

Parameters:

oSizeROI ROI size.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.55 NppStatus nppiReductionGetBufferHostSize_8u_C4R (NppiSize oSizeROI, int * hpBufferSize)

Scratch-buffer size for nppiSum_8u_C4R.

Parameters:

oSizeROI ROI size.

hpBufferSize Host pointer where required buffer size is returned.

Returns:

Error Code.

7.9.2.56 NppStatus nppiSqrIntegral_8u32s32f_C1R (Npp8u * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, Npp32f * pSqr, int nSqrStep, NppiSize oSrcROI, Npp32s val, Npp32f valSqr, Npp32s integralImageNewHeight)

SqrIntegral Transforms an image to integral and integral of pixel squares representation.

This function assumes that the integral and integral of squares images.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

pSqr Destination-Image Pointer.

nSqrStep Destination-Image Line Step.

oSrcROI Region-of-Interest (ROI).

val The value to add to pDst image pixels

valSqr The value to add to pSqr image pixels

integralImageNewHeight Extended height of output surfaces (needed by transpose in primitive)

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.2.57 `NppStatus nppiSum_8u64s_C1R (const Npp8u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp8u * pDeviceBuffer, Npp64s * pSum)`

8-bit unsigned image sum with 64-bit long long result.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pDeviceBuffer Pointer to the required device memory allocation, Scratch Buffer and Host Pointer

**pSum* Contains computed sum.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.2.58 `NppStatus nppiSum_8u64s_C4R (const Npp8u * pSrc, int nSrcStep, NppiSize oSizeROI, Npp8u * pDeviceBuffer, Npp64s aSum[4])`

4 channel 8-bit unsigned image sum with 64-bit long long result.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pDeviceBuffer Pointer to the required device memory allocation, Scratch Buffer and Host Pointer

aSum Array contains computed sum for each channel.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.9.2.59 NppStatus nppiSum_8u_C1R (const Npp8u * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp8u * *pDeviceBuffer*, Npp64f * *pSum*)

8-bit unsigned image sum with 64-bit double precision result.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pDeviceBuffer Pointer to the required device memory allocation, [Scratch Buffer and Host Pointer](#)

**pSum* Contains computed sum.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.9.2.60 NppStatus nppiSum_8u_C4R (const Npp8u * *pSrc*, int *nSrcStep*, NppiSize *oSizeROI*, Npp8u * *pDeviceBuffer*, Npp64f *aSum*[4])

4 channel 8-bit unsigned image sum with 64-bit double precision result.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pDeviceBuffer Pointer to the required device memory allocation, [Scratch Buffer and Host Pointer](#)

aSum Array contains computed sum for each channel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10 Filtering Functions

Linear and non-linear image filtering functions.

1D Linear Filter

1D mask Linear Convolution Filter, with rescaling, for 8 bit images.

- `NppStatus nppiFilterColumn_8u_C1R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oROI, const `Npp32s` *pKernel, `Npp32s` nMaskSize, `Npp32s` nAnchor, `Npp32s` nDivisor)
8-bit unsigned 1D (column) image convolution.
- `NppStatus nppiFilterColumn_8u_C4R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oROI, const `Npp32s` *pKernel, `Npp32s` nMaskSize, `Npp32s` nAnchor, `Npp32s` nDivisor)
4 channel 8-bit unsigned 1D (column) image convolution.
- `NppStatus nppiFilterRow_8u_C1R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oROI, const `Npp32s` *pKernel, `Npp32s` nMaskSize, `Npp32s` nAnchor, `Npp32s` nDivisor)
8-bit unsigned 1D (row) image convolution.
- `NppStatus nppiFilterRow_8u_C4R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oROI, const `Npp32s` *pKernel, `Npp32s` nMaskSize, `Npp32s` nAnchor, `Npp32s` nDivisor)
4 channel 8-bit unsigned 1D (row) image convolution.

1D Window Sum

1D mask Window Sum for 8 bit images.

- `NppStatus nppiSumWindowColumn_8u32f_C1R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp32f` *pDst, `Npp32s` nDstStep, `NppiSize` oROI, `Npp32s` nMaskSize, `Npp32s` nAnchor)
8-bit unsigned 1D (column) sum to 32f.
- `NppStatus nppiSumWindowRow_8u32f_C1R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp32f` *pDst, `Npp32s` nDstStep, `NppiSize` oROI, `Npp32s` nMaskSize, `Npp32s` nAnchor)
8-bit unsigned 1D (row) sum to 32f.

Convolution (2D Masks)

General purpose 2D convolution filters.

- `NppStatus nppiFilter_8u_C1R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oSizeROI, const `Npp32s` *pKernel, `NppiSize` oKernelSize, `NppiPoint` oAnchor, `Npp32s` nDivisor)

8-bit unsigned convolution filter.

- `NppStatus nppiFilter_8u_C4R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oSizeROI, const `Npp32s` *pKernel, `NppiSize` oKernelSize, `NppiPoint` oAnchor, `Npp32s` nDivisor)

4 channel 8-bit unsigned convolution filter.

2D Linear Fixed Filters

2D linear fixed filters for 8 bit images.

- `NppStatus nppiFilterBox_8u_C1R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oSizeROI, `NppiSize` oMaskSize, `NppiPoint` oAnchor)

8-bit unsigned box filter.

- `NppStatus nppiFilterBox_8u_C4R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oSizeROI, `NppiSize` oMaskSize, `NppiPoint` oAnchor)

4 channel 8-bit unsigned box filter.

Image Rank Filters

Min, Median, and Max image filters.

- `NppStatus nppiFilterMax_8u_C1R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oSizeROI, `NppiSize` oMaskSize, `NppiPoint` oAnchor)

8-bit unsigned maximum filter.

- `NppStatus nppiFilterMax_8u_C4R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oSizeROI, `NppiSize` oMaskSize, `NppiPoint` oAnchor)

4 channel 8-bit unsigned maximum filter.

- `NppStatus nppiFilterMin_8u_C1R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oSizeROI, `NppiSize` oMaskSize, `NppiPoint` oAnchor)

8-bit unsigned minimum filter.

- `NppStatus nppiFilterMin_8u_C4R` (const `Npp8u` *pSrc, `Npp32s` nSrcStep, `Npp8u` *pDst, `Npp32s` nDstStep, `NppiSize` oSizeROI, `NppiSize` oMaskSize, `NppiPoint` oAnchor)

4 channel 8-bit unsigned minimum filter.

7.10.1 Detailed Description

Linear and non-linear image filtering functions.

7.10.2 Function Documentation

7.10.2.1 NppStatus nppiFilter_8u_C1R (const Npp8u * *pSrc*, Npp32s *nSrcStep*, Npp8u * *pDst*, Npp32s *nDstStep*, NppiSize *oSizeROI*, const Npp32s * *pKernel*, NppiSize *oKernelSize*, NppiPoint *oAnchor*, Npp32s *nDivisor*)

8-bit unsigned convolution filter.

Pixels under the mask are multiplied by the respective weights in the mask and the results are summed. Before writing the result pixel the sum is scaled back via division by *nDivisor*.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pKernel Pointer to the start address of the kernel coefficient array. Coefficients are expected to be stored in reverse order.

oKernelSize Width and Height of the rectangular kernel.

oAnchor X and Y offsets of the kernel origin frame of reference w.r.t the source pixel.

nDivisor The factor by which the convolved summation from the Filter operation should be divided. If equal to the sum of coefficients, this will keep the maximum result value within full scale.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.2 NppStatus nppiFilter_8u_C4R (const Npp8u * *pSrc*, Npp32s *nSrcStep*, Npp8u * *pDst*, Npp32s *nDstStep*, NppiSize *oSizeROI*, const Npp32s * *pKernel*, NppiSize *oKernelSize*, NppiPoint *oAnchor*, Npp32s *nDivisor*)

4 channel 8-bit unsigned convolution filter.

Pixels under the mask are multiplied by the respective weights in the mask and the results are summed. Before writing the result pixel the sum is scaled back via division by *nDivisor*.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pKernel Pointer to the start address of the kernel coefficient array. Coefficients are expected to be stored in reverse order.

oKernelSize Width and Height of the rectangular kernel.

oAnchor X and Y offsets of the kernel origin frame of reference w.r.t the source pixel.

nDivisor The factor by which the convolved summation from the Filter operation should be divided. If equal to the sum of coefficients, this will keep the maximum result value within full scale.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.3 `NppStatus nppiFilterBox_8u_C1R (const Npp8u *pSrc, Npp32s nSrcStep, Npp8u *pDst, Npp32s nDstStep, NppiSize oSizeROI, NppiSize oMaskSize, NppiPoint oAnchor)`

8-bit unsigned box filter.

Computes the average pixel values of the pixels under a rectangular mask.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

oMaskSize Width and Height of the neighborhood region for the local Avg operation.

oAnchor X and Y offsets of the kernel origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.4 `NppStatus nppiFilterBox_8u_C4R (const Npp8u *pSrc, Npp32s nSrcStep, Npp8u *pDst, Npp32s nDstStep, NppiSize oSizeROI, NppiSize oMaskSize, NppiPoint oAnchor)`

4 channel 8-bit unsigned box filter.

Computes the average pixel values of the pixels under a rectangular mask.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

oMaskSize Width and Height of the neighborhood region for the local Avg operation.

oAnchor X and Y offsets of the kernel origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.5 NppStatus nppiFilterColumn_8u_C1R (const Npp8u * *pSrc*, Npp32s *nSrcStep*, Npp8u * *pDst*, Npp32s *nDstStep*, NppiSize *oROI*, const Npp32s * *pKernel*, Npp32s *nMaskSize*, Npp32s *nAnchor*, Npp32s *nDivisor*)

8-bit unsigned 1D (column) image convolution.

Apply convolution filter with user specified 1D column of weights. Result pixel is equal to the sum of the products between the kernel coefficients (*pKernel* array) and corresponding neighboring column pixel values in the source image defined by *nKernelDim* and *nAnchorY*, divided by *nDivisor*.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oROI [Region-of-Interest \(ROI\)](#).

pKernel Pointer to the start address of the kernel coefficient array. Coefficients are expected to be stored in reverse order.

nMaskSize Length of the linear kernel array.

nAnchor Y offset of the kernel origin frame of reference w.r.t the source pixel.

nDivisor The factor by which the convolved summation from the Filter operation should be divided. If equal to the sum of coefficients, this will keep the maximum result value within full scale.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.6 NppStatus nppiFilterColumn_8u_C4R (const Npp8u * *pSrc*, Npp32s *nSrcStep*, Npp8u * *pDst*, Npp32s *nDstStep*, NppiSize *oROI*, const Npp32s * *pKernel*, Npp32s *nMaskSize*, Npp32s *nAnchor*, Npp32s *nDivisor*)

4 channel 8-bit unsigned 1D (column) image convolution.

Apply convolution filter with user specified 1D column of weights. Result pixel is equal to the sum of the products between the kernel coefficients (*pKernel* array) and corresponding neighboring column pixel values in the source image defined by *nKernelDim* and *nAnchorY*, divided by *nDivisor*.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oROI [Region-of-Interest \(ROI\)](#).

pKernel Pointer to the start address of the kernel coefficient array. Coefficients are expected to be stored in reverse order.

nMaskSize Length of the linear kernel array.

nAnchor Y offset of the kernel origin frame of reference w.r.t the source pixel.

nDivisor The factor by which the convolved summation from the Filter operation should be divided. If equal to the sum of coefficients, this will keep the maximum result value within full scale.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.7 `NppStatus nppiFilterMax_8u_C1R (const Npp8u * pSrc, Npp32s nSrcStep, Npp8u * pDst, Npp32s nDstStep, NppiSize oSizeROI, NppiSize oMaskSize, NppiPoint oAnchor)`

8-bit unsigned maximum filter.

Result pixel value is the maximum of pixel values under the rectangular mask region.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

oMaskSize Width and Height of the neighborhood region for the local Max operation.

oAnchor X and Y offsets of the kernel origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.8 `NppStatus nppiFilterMax_8u_C4R (const Npp8u * pSrc, Npp32s nSrcStep, Npp8u * pDst, Npp32s nDstStep, NppiSize oSizeROI, NppiSize oMaskSize, NppiPoint oAnchor)`

4 channel 8-bit unsigned maximum filter.

Result pixel value is the maximum of pixel values under the rectangular mask region.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

oMaskSize Width and Height of the neighborhood region for the local Max operation.

oAnchor X and Y offsets of the kernel origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.9 NppStatus nppiFilterMin_8u_C1R (const Npp8u * *pSrc*, Npp32s *nSrcStep*, Npp8u * *pDst*, Npp32s *nDstStep*, NppiSize *oSizeROI*, NppiSize *oMaskSize*, NppiPoint *oAnchor*)

8-bit unsigned minimum filter.

Result pixel value is the minimum of pixel values under the rectangular mask region.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

oMaskSize Width and Height of the neighborhood region for the local Max operation.

oAnchor X and Y offsets of the kernel origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.10 NppStatus nppiFilterMin_8u_C4R (const Npp8u * *pSrc*, Npp32s *nSrcStep*, Npp8u * *pDst*, Npp32s *nDstStep*, NppiSize *oSizeROI*, NppiSize *oMaskSize*, NppiPoint *oAnchor*)

4 channel 8-bit unsigned minimum filter.

Result pixel value is the minimum of pixel values under the rectangular mask region.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

oMaskSize Width and Height of the neighborhood region for the local Max operation.

oAnchor X and Y offsets of the kernel origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.11 NppStatus nppiFilterRow_8u_C1R (const Npp8u * *pSrc*, Npp32s *nSrcStep*, Npp8u * *pDst*, Npp32s *nDstStep*, NppiSize *oROI*, const Npp32s * *pKernel*, Npp32s *nMaskSize*, Npp32s *nAnchor*, Npp32s *nDivisor*)

8-bit unsigned 1D (row) image convolution.

Apply general linear Row convolution filter, with rescaling, in a 1D mask region around each source pixel for 1-channel 8 bit/pixel images. Result pixel is equal to the sum of the products between the kernel coefficients (*pKernel* array) and corresponding neighboring row pixel values in the source image defined by *iKernelDim* and *iAnchorX*, divided by *iDivisor*.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oROI [Region-of-Interest \(ROI\)](#).

pKernel Pointer to the start address of the kernel coefficient array. Coefficients are expected to be stored in reverse order.

nMaskSize Length of the linear kernel array.

nAnchor X offset of the kernel origin frame of reference w.r.t the source pixel.

nDivisor The factor by which the convolved summation from the Filter operation should be divided. If equal to the sum of coefficients, this will keep the maximum result value within full scale.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.12 NppStatus nppiFilterRow_8u_C4R (const Npp8u * pSrc, Npp32s nSrcStep, Npp8u * pDst, Npp32s nDstStep, NppiSize oROI, const Npp32s * pKernel, Npp32s nMaskSize, Npp32s nAnchor, Npp32s nDivisor)

4 channel 8-bit unsigned 1D (row) image convolution.

Apply general linear Row convolution filter, with rescaling, in a 1D mask region around each source pixel for 1-channel 8 bit/pixel images. Result pixel is equal to the sum of the products between the kernel coefficients (pKernel array) and corresponding neighboring row pixel values in the source image defined by iKernelDim and iAnchorX, divided by iDivisor.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oROI [Region-of-Interest \(ROI\)](#).

pKernel Pointer to the start address of the kernel coefficient array. Coefficients are expected to be stored in reverse order.

nMaskSize Length of the linear kernel array.

nAnchor X offset of the kernel origin frame of reference w.r.t the source pixel.

nDivisor The factor by which the convolved summation from the Filter operation should be divided. If equal to the sum of coefficients, this will keep the maximum result value within full scale.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.13 NppStatus nppiSumWindowColumn_8u32f_C1R (const Npp8u * *pSrc*, Npp32s *nSrcStep*, Npp32f * *pDst*, Npp32s *nDstStep*, NppiSize *oROI*, Npp32s *nMaskSize*, Npp32s *nAnchor*)

8-bit unsigned 1D (column) sum to 32f.

Apply Column Window Summation filter over a 1D mask region around each source pixel for 1-channel 8 bit/pixel input images with 32-bit floating point output. Result 32-bit floating point pixel is equal to the sum of the corresponding and neighboring column pixel values in a mask region of the source image defined by *nMaskSize* and *nAnchor*.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oROI Region-of-Interest (ROI).

nMaskSize Length of the linear kernel array.

nAnchor Y offset of the kernel origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.10.2.14 NppStatus nppiSumWindowRow_8u32f_C1R (const Npp8u * *pSrc*, Npp32s *nSrcStep*, Npp32f * *pDst*, Npp32s *nDstStep*, NppiSize *oROI*, Npp32s *nMaskSize*, Npp32s *nAnchor*)

8-bit unsigned 1D (row) sum to 32f.

Apply Row Window Summation filter over a 1D mask region around each source pixel for 1-channel 8-bit pixel input images with 32-bit floating point output. Result 32-bit floating point pixel is equal to the sum of the corresponding and neighboring row pixel values in a mask region of the source image defined by *iKernelDim* and *iAnchorX*.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oROI Region-of-Interest (ROI).

nMaskSize Length of the linear kernel array.

nAnchor X offset of the kernel origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11 Morphological Operations

Morphological image operations.

Functions

- **NppStatus nppiDilate_8u_C1R** (const **Npp8u** *pSrc, **Npp32s** nSrcStep, **Npp8u** *pDst, **Npp32s** nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, **NppiSize** oMaskSize, **NppiPoint** oAnchor)
8-bit unsigned image dilation.
- **NppStatus nppiDilate_8u_C4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, **NppiSize** oMaskSize, **NppiPoint** oAnchor)
4 channel 8-bit unsigned image dilation.
- **NppStatus nppiErode_8u_C1R** (const **Npp8u** *pSrc, **Npp32s** nSrcStep, **Npp8u** *pDst, **Npp32s** nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, **NppiSize** oMaskSize, **NppiPoint** oAnchor)
8-bit unsigned image erosion.
- **NppStatus nppiErode_8u_C4R** (const **Npp8u** *pSrc, **Npp32s** nSrcStep, **Npp8u** *pDst, **Npp32s** nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, **NppiSize** oMaskSize, **NppiPoint** oAnchor)
4 channel 8-bit unsigned image erosion.

7.11.1 Detailed Description

Morphological image operations.

7.11.2 Function Documentation

7.11.2.1 NppStatus nppiDilate_8u_C1R (const **Npp8u** *pSrc, **Npp32s** nSrcStep, **Npp8u** *pDst, **Npp32s** nDstStep, **NppiSize** oSizeROI, const **Npp8u** *pMask, **NppiSize** oMaskSize, **NppiPoint** oAnchor)

8-bit unsigned image dilation.

Dilation computes the output pixel as the maximum pixel value of the pixels under the mask. Pixels whose corresponding mask values are zero do not participate in the maximum search.

Parameters:

- pSrc** Source-Image Pointer.
- nSrcStep** Source-Image Line Step.
- pDst** Destination-Image Pointer.
- nDstStep** Destination-Image Line Step.
- oSizeROI** Region-of-Interest (ROI).
- pMask** Pointer to the start address of the mask array
- oMaskSize** Width and Height mask array.
- oAnchor** X and Y offsets of the mask origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.2.2 `NppStatus nppiDilate_8u_C4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oSizeROI, const Npp8u * pMask, NppiSize oMaskSize, NppiPoint oAnchor)`

4 channel 8-bit unsigned image dilation.

Dilation computes the output pixel as the maximum pixel value of the pixels under the mask. Pixels whose corresponding mask values are zero do not participate in the maximum search.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMask Pointer to the start address of the mask array

oMaskSize Width and Height mask array.

oAnchor X and Y offsets of the mask origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.2.3 `NppStatus nppiErode_8u_C1R (const Npp8u * pSrc, Npp32s nSrcStep, Npp8u * pDst, Npp32s nDstStep, NppiSize oSizeROI, const Npp8u * pMask, NppiSize oMaskSize, NppiPoint oAnchor)`

8-bit unsigned image erosion.

Erosion computes the output pixel as the minimum pixel value of the pixels under the mask. Pixels whose corresponding mask values are zero do not participate in the maximum search.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMask Pointer to the start address of the mask array

oMaskSize Width and Height mask array.

oAnchor X and Y offsets of the mask origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.11.2.4 NppStatus nppiErode_8u_C4R (const Npp8u * *pSrc*, Npp32s *nSrcStep*, Npp8u * *pDst*, Npp32s *nDstStep*, NppiSize *oSizeROI*, const Npp8u * *pMask*, NppiSize *oMaskSize*, NppiPoint *oAnchor*)

4 channel 8-bit unsigned image erosion.

Erosion computes the output pixel as the minimum pixel value of the pixels under the mask. Pixels whose corresponding mask values are zero do not participate in the maximum search.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

pMask Pointer to the start address of the mask array

oMaskSize Width and Height mask array.

oAnchor X and Y offsets of the mask origin frame of reference w.r.t the source pixel.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12 Image Linear Transforms

Linear image transformations.

Functions

- **NppStatus nppiMagnitude_32fc32f_C1R** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
32-bit floating point complex to 32-bit floating point magnitude.
- **NppStatus nppiMagnitudeSqr_32fc32f_C1R** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)
32-bit floating point complex to 32-bit floating point squared magnitude.

7.12.1 Detailed Description

Linear image transformations.

7.12.2 Function Documentation

7.12.2.1 **NppStatus nppiMagnitude_32fc32f_C1R** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

32-bit floating point complex to 32-bit floating point magnitude.

Converts complex-number pixel image to single channel image computing the result pixels as the magnitude of the complex values.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.12.2.2 **NppStatus nppiMagnitudeSqr_32fc32f_C1R** (const **Npp32fc** *pSrc, int nSrcStep, **Npp32f** *pDst, int nDstStep, **NppiSize** oSizeROI)

32-bit floating point complex to 32-bit floating point squared magnitude.

Converts complex-number pixel image to single channel image computing the result pixels as the squared magnitude of the complex values.

The squared magnitude is an intermediate result of magnitude computation and can thus be computed faster than actual magnitude. If magnitudes are required for sorting/comparing only, using this function instead of `nppiMagnitude_32fc32f_C1R` can be a worthwhile performance optimization.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.13 Compression

Image compression primitives.

Functions

- **NppStatus** **nppiQuantFwdRawTableInit_JPEG_8u** (**Npp8u** *hpQuantRawTable, int nQualityFactor)
Apply quality factor to raw 8-bit quantization table.
- **NppStatus** **nppiQuantFwdTableInit_JPEG_8u16u** (const **Npp8u** *hpQuantRawTable, **Npp16u** *hpQuantFwdRawTable)
*Initializes a quantization table for **nppiDCTQuantFwd8x8LS_JPEG_8u16s_C1R()**.*
- **NppStatus** **nppiQuantInvTableInit_JPEG_8u16u** (const **Npp8u** *hpQuantRawTable, **Npp16u** *hpQuantFwdRawTable)
*Initializes a quantization table for **nppiDCTQuantInv8x8LS_JPEG_16s8u_C1R()**.*
- **NppStatus** **nppiDCTQuantFwd8x8LS_JPEG_8u16s_C1R** (const **Npp8u** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, const **Npp16u** *pQuantFwdTable, **NppiSize** oSizeROI)
Forward DCT, quantization and level shift part of the JPEG encoding.
- **NppStatus** **nppiDCTQuantInv8x8LS_JPEG_16s8u_C1R** (const **Npp16s** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, const **Npp16u** *pQuantInvTable, **NppiSize** oSizeROI)
Inverse DCT, de-quantization and level shift part of the JPEG decoding.

7.13.1 Detailed Description

Image compression primitives.

The JPEG standard defines a flow of level shift, DCT and quantization for forward JPEG transform and inverse level shift, IDCT and de-quantization for inverse JPEG transform. This group has the functions for both forward and inverse functions.

7.13.2 Function Documentation

7.13.2.1 **NppStatus** **nppiDCTQuantFwd8x8LS_JPEG_8u16s_C1R** (const **Npp8u** *pSrc, int nSrcStep, **Npp16s** *pDst, int nDstStep, const **Npp16u** *pQuantFwdTable, **NppiSize** oSizeROI)

Forward DCT, quantization and level shift part of the JPEG encoding.

Input is expected in 8x8 macro blocks and output is expected to be in 64x1 macro blocks.

Parameters:

- pSrc** Source-Image Pointer.
- nSrcStep** Source-Image Line Step.
- pDst** Destination-Image Pointer.

nDstStep Destination-Image Line Step.

pQuantFwdTable Forward quantization tables for JPEG encoding created using `nppiQuantInvTableInit_JPEG_8u16u()`.

oSizeROI Region-of-Interest (ROI).

Returns:

Error codes:

- `NPP_SIZE_ERROR` For negative input height/width or not a multiple of 8 width/height.
- `NPP_STEP_ERROR` If input image width is not multiple of 8 or does not match ROI.
- `NPP_NULL_POINTER_ERROR` If the destination pointer is 0.

7.13.2.2 `NppStatus nppiDCTQuantInv8x8LS_JPEG_16s8u_C1R (const Npp16s * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, const Npp16u * pQuantInvTable, NppiSize oSizeROI)`

Inverse DCT, de-quantization and level shift part of the JPEG decoding.

Input is expected in 64x1 macro blocks and output is expected to be in 8x8 macro blocks.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

pQuantInvTable Inverse quantization tables for JPEG decoding created using `nppiQuantInvTableInit_JPEG_8u16u()`.

oSizeROI Region-of-Interest (ROI).

Returns:

Error codes:

- `NPP_SIZE_ERROR` For negative input height/width or not a multiple of 8 width/height.
- `NPP_STEP_ERROR` If input image width is not multiple of 8 or does not match ROI.
- `NPP_NULL_POINTER_ERROR` If the destination pointer is 0.

7.13.2.3 `NppStatus nppiQuantFwdRawTableInit_JPEG_8u (Npp8u * hpQuantRawTable, int nQualityFactor)`

Apply quality factor to raw 8-bit quantization table.

This is effectively and in-place method that modifies a given raw quantization table based on a quality factor. Note that this method is a host method and that the pointer to the raw quantization table is a host pointer.

Parameters:

hpQuantRawTable Raw quantization table.

nQualityFactor Quality factor for the table. Range is [1:100].

Returns:

Error code: [NPP_NULL_POINTER_ERROR](#) is returned if hpQuantRawTable is 0.

7.13.2.4 NppStatus nppiQuantFwdTableInit_JPEG_8u16u (const Npp8u * hpQuantRawTable, Npp16u * hpQuantFwdRawTable)

Initializes a quantization table for [nppiDCTQuantFwd8x8LS_JPEG_8u16s_C1R\(\)](#).

The method creates a 16-bit version of the raw table and converts the data order from zigzag layout to original row-order layout since raw quantization tables are typically stored in zigzag format.

This method is a host method. It consumes and produces host data. I.e. the pointers passed to this function must be host pointers. The resulting table needs to be transferred to device memory in order to be used with [nppiDCTQuantFwd8x8LS_JPEG_8u16s_C1R\(\)](#) function.

Parameters:

hpQuantRawTable Host pointer to raw quantization table as returned by [nppiQuantFwdRawTableInit_JPEG_8u\(\)](#). The raw quantization table is assumed to be in zigzag order.

hpQuantFwdRawTable Forward quantization table for use with [nppiDCTQuantFwd8x8LS_JPEG_8u16s_C1R\(\)](#).

Returns:

Error code: [NPP_NULL_POINTER_ERROR](#) pQuantRawTable is 0.

7.13.2.5 NppStatus nppiQuantInvTableInit_JPEG_8u16u (const Npp8u * hpQuantRawTable, Npp16u * hpQuantFwdRawTable)

Initializes a quantization table for [nppiDCTQuantInv8x8LS_JPEG_16s8u_C1R\(\)](#).

The [nppiDCTQuantFwd8x8LS_JPEG_8u16s_C1R\(\)](#) method uses a quantization table in a 16-bit format allowing for faster processing. In addition it converts the data order from zigzag layout to original row-order layout. Typically raw quantization tables are stored in zigzag format.

This method is a host method and consumes and produces host data. I.e. the pointers passed to this function must be host pointers. The resulting table needs to be transferred to device memory in order to be used with [nppiDCTQuantFwd8x8LS_JPEG_8u16s_C1R\(\)](#) function.

Parameters:

hpQuantRawTable Raw quantization table.

hpQuantFwdRawTable Inverse quantization table.

Returns:

[NPP_NULL_POINTER_ERROR](#) pQuantRawTable or pQuantFwdRawTable is 0.

7.14 Geometric Transforms

Routines manipulating an image's geometry.

Resize

Resizes 8 bit images.

Handles C1 and C4 images.

- `NppStatus nppiResize_8u_C1R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp8u` *pDst, int nDstStep, `NppiSize` dstROISize, double xFactor, double yFactor, int eInterpolation)

8-bit unsigned image resize.

- `NppStatus nppiResize_8u_C4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp8u` *pDst, int nDstStep, `NppiSize` dstROISize, double xFactor, double yFactor, int eInterpolation)

4 channel 8-bit unsigned image resize.

Rotate

Rotates an image around the origin (0,0) and then shifts it.

- `NppStatus nppiGetRotateQuad` (`NppiRect` oSrcROI, double aQuad[4][2], double nAngle, double nShiftX, double nShiftY)

Compute shape of rotated image.

- `NppStatus nppiGetRotateBound` (`NppiRect` oSrcROI, double aBoundingBox[2][2], double nAngle, double nShiftX, double nShiftY)

Compute bounding-box of rotated image.

- `NppStatus nppiRotate_8u_C1R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp8u` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

8-bit unsigned image rotate.

- `NppStatus nppiRotate_8u_C3R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp8u` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

3 channel 8-bit unsigned image rotate.

- `NppStatus nppiRotate_8u_C4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp8u` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

4 channel 8-bit unsigned image rotate.

- `NppStatus nppiRotate_8u_AC4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp8u` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

4 channel 8-bit unsigned image rotate ignoring alpha channel.

- `NppStatus nppiRotate_16u_C1R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp16u` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

16-bit unsigned image rotate.

- `NppStatus nppiRotate_16u_C3R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp16u` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

3 channel 16-bit unsigned image rotate.

- `NppStatus nppiRotate_16u_C4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp16u` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

4 channel 16-bit unsigned image rotate.

- `NppStatus nppiRotate_16u_AC4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp16u` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

4 channel 16-bit unsigned image rotate ignoring alpha channel.

- `NppStatus nppiRotate_32f_C1R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp32f` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

32-bit float image rotate.

- `NppStatus nppiRotate_32f_C3R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp32f` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

3 channel 32-bit float image rotate.

- `NppStatus nppiRotate_32f_C4R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp32f` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

4 channel 32-bit float image rotate.

- `NppStatus nppiRotate_32f_AC4R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` oSrcROI, `Npp32f` *pDst, int nDstStep, `NppiRect` oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)

4 channel 32-bit float image rotate ignoring alpha channel.

Mirror

Mirrors images horizontally, vertically and diagonally.

- `NppStatus nppiMirror_8u_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)

8-bit unsigned image mirror.

- `NppStatus nppiMirror_8u_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
3 channel 8-bit unsigned image mirror.
- `NppStatus nppiMirror_8u_C4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
4 channel 8-bit unsigned image mirror.
- `NppStatus nppiMirror_8u_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
4 channel 8-bit unsigned image mirror not affecting alpha.
- `NppStatus nppiMirror_16u_C1R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
16-bit unsigned image mirror.
- `NppStatus nppiMirror_16u_C3R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
3 channel 16-bit unsigned image mirror.
- `NppStatus nppiMirror_16u_C4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
4 channel 16-bit unsigned image mirror.
- `NppStatus nppiMirror_16u_AC4R` (const `Npp16u` *pSrc, int nSrcStep, `Npp16u` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
4 channel 16-bit unsigned image mirror not affecting alpha.
- `NppStatus nppiMirror_32s_C1R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
32-bit image mirror.
- `NppStatus nppiMirror_32s_C3R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
3 channel 32-bit image mirror.
- `NppStatus nppiMirror_32s_C4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
4 channel 32-bit image mirror.
- `NppStatus nppiMirror_32s_AC4R` (const `Npp32s` *pSrc, int nSrcStep, `Npp32s` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
4 channel 32-bit image mirror not affecting alpha.
- `NppStatus nppiMirror_32f_C1R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
32-bit float image mirror.
- `NppStatus nppiMirror_32f_C3R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
3 channel 32-bit float image mirror.

- `NppStatus nppiMirror_32f_C4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
4 channel 32-bit float image mirror.
- `NppStatus nppiMirror_32f_AC4R` (const `Npp32f` *pSrc, int nSrcStep, `Npp32f` *pDst, int nDstStep, `NppiSize` oROI, `NppiAxis` flip)
4 channel 32-bit float image mirror not affecting alpha.

Affine warping, affine transform calculation

Affine warping of an image is the transform of image pixel positions, defined by the following formulas:

$$X_{new} = C_{00} * x + C_{01} * y + C_{02} \quad Y_{new} = C_{10} * x + C_{11} * y + C_{12} \quad C = \begin{bmatrix} C_{00} & C_{01} & C_{02} \\ C_{10} & C_{11} & C_{12} \end{bmatrix}$$

That is, any pixel with coordinates (X_{new}, Y_{new}) in the transformed image is sourced from coordinates (x, y) in the original image.

The mapping C is completely specified by 6 values $C_{ij}, i = \overline{0,1}, j = \overline{0,2}$. The transform maps parallel lines to parallel lines and preserves ratios of distances of points to lines. Implementation specific properties are discussed in each function's documentation.

- `NppStatus nppiGetAffineTransform` (`NppiRect` srcRoi, const double quad[4][2], double coeffs[2][3])
Calculates affine transform coefficients given source rectangular ROI and its destination quadrangle projection.
- `NppStatus nppiGetAffineQuad` (`NppiRect` srcRoi, double quad[4][2], const double coeffs[2][3])
Calculates affine transform projection of given source rectangular ROI.
- `NppStatus nppiGetAffineBound` (`NppiRect` srcRoi, double bound[2][2], const double coeffs[2][3])
Calculates bounding box of the affine transform projection of the given source rectangular ROI.
- `NppStatus nppiWarpAffine_8u_C1R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)
Affine transform of an image (8bit unsigned integer, single channel).
- `NppStatus nppiWarpAffine_8u_C3R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)
Affine transform of an image (8bit unsigned integer, three channels).
- `NppStatus nppiWarpAffine_8u_C4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)
Affine transform of an image (8bit unsigned integer, four channels).
- `NppStatus nppiWarpAffine_8u_AC4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (8bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpAffine_8u_P3R` (const `Npp8u *pSrc[3]`, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u *pDst[3]`, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (8bit unsigned integer, three planes).

- `NppStatus nppiWarpAffine_8u_P4R` (const `Npp8u *pSrc[4]`, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u *pDst[4]`, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (8bit unsigned integer, four planes).

- `NppStatus nppiWarpAffineBack_8u_C1R` (const `Npp8u *pSrc`, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u *pDst`, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (8bit unsigned integer, single channel).

- `NppStatus nppiWarpAffineBack_8u_C3R` (const `Npp8u *pSrc`, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u *pDst`, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (8bit unsigned integer, three channels).

- `NppStatus nppiWarpAffineBack_8u_C4R` (const `Npp8u *pSrc`, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u *pDst`, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (8bit unsigned integer, four channels).

- `NppStatus nppiWarpAffineBack_8u_AC4R` (const `Npp8u *pSrc`, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u *pDst`, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (8bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpAffineBack_8u_P3R` (const `Npp8u *pSrc[3]`, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u *pDst[3]`, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (8bit unsigned integer, three planes).

- `NppStatus nppiWarpAffineBack_8u_P4R` (const `Npp8u *pSrc[4]`, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u *pDst[4]`, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (8bit unsigned integer, four planes).

- `NppStatus nppiWarpAffineQuad_8u_C1R` (const `Npp8u *pSrc`, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u *pDst`, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (8bit unsigned integer, single channel).

- `NppStatus nppiWarpAffineQuad_8u_C3R` (const `Npp8u *pSrc`, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u *pDst`, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (8bit unsigned integer, three channels).

- `NppStatus nppiWarpAffineQuad_8u_C4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (8bit unsigned integer, four channels).

- `NppStatus nppiWarpAffineQuad_8u_AC4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (8bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpAffineQuad_8u_P3R` (const `Npp8u` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (8bit unsigned integer, three planes).

- `NppStatus nppiWarpAffineQuad_8u_P4R` (const `Npp8u` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (8bit unsigned integer, four planes).

- `NppStatus nppiWarpAffine_16u_C1R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (16bit unsigned integer, single channel).

- `NppStatus nppiWarpAffine_16u_C3R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (16bit unsigned integer, three channels).

- `NppStatus nppiWarpAffine_16u_C4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (16bit unsigned integer, four channels).

- `NppStatus nppiWarpAffine_16u_AC4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (16bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpAffine_16u_P3R` (const `Npp16u` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (16bit unsigned integer, three planes).

- `NppStatus nppiWarpAffine_16u_P4R` (const `Npp16u` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (16bit unsigned integer, four planes).

- `NppStatus nppiWarpAffineBack_16u_C1R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (16bit unsigned integer, single channel).

- `NppStatus nppiWarpAffineBack_16u_C3R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (16bit unsigned integer, three channels).

- `NppStatus nppiWarpAffineBack_16u_C4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (16bit unsigned integer, four channels).

- `NppStatus nppiWarpAffineBack_16u_AC4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (16bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpAffineBack_16u_P3R` (const `Npp16u` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (16bit unsigned integer, three planes).

- `NppStatus nppiWarpAffineBack_16u_P4R` (const `Npp16u` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (16bit unsigned integer, four planes).

- `NppStatus nppiWarpAffineQuad_16u_C1R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (16bit unsigned integer, single channel).

- `NppStatus nppiWarpAffineQuad_16u_C3R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (16bit unsigned integer, three channels).

- `NppStatus nppiWarpAffineQuad_16u_C4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (16bit unsigned integer, four channels).

- `NppStatus nppiWarpAffineQuad_16u_AC4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (16bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpAffineQuad_16u_P3R` (const `Npp16u` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (16bit unsigned integer, three planes).

- `NppStatus nppiWarpAffineQuad_16u_P4R` (const `Npp16u` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (16bit unsigned integer, four planes).

- `NppStatus nppiWarpAffine_32f_C1R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit float, single channel).

- `NppStatus nppiWarpAffine_32f_C3R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit float, three channels).

- `NppStatus nppiWarpAffine_32f_C4R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit float, four channels).

- `NppStatus nppiWarpAffine_32f_AC4R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit float, four channels RGBA).

- `NppStatus nppiWarpAffine_32f_P3R` (const `Npp32f` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32f` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit float, three planes).

- `NppStatus nppiWarpAffine_32f_P4R` (const `Npp32f` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32f` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit float, four planes).

- `NppStatus nppiWarpAffineBack_32f_C1R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit float, single channel).

- `NppStatus nppiWarpAffineBack_32f_C3R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit float, three channels).

- **NppStatus** **nppiWarpAffineBack_32f_C4R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit float, four channels).

- **NppStatus** **nppiWarpAffineBack_32f_AC4R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit float, four channels RGBA).

- **NppStatus** **nppiWarpAffineBack_32f_P3R** (const **Npp32f** *pSrc[3], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst[3], int nDstStep, **NppiRect** dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit float, three planes).

- **NppStatus** **nppiWarpAffineBack_32f_P4R** (const **Npp32f** *pSrc[4], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst[4], int nDstStep, **NppiRect** dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit float, four planes).

- **NppStatus** **nppiWarpAffineQuad_32f_C1R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit float, single channel).

- **NppStatus** **nppiWarpAffineQuad_32f_C3R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit float, three channels).

- **NppStatus** **nppiWarpAffineQuad_32f_C4R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit float, four channels).

- **NppStatus** **nppiWarpAffineQuad_32f_AC4R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit float, four channels RGBA).

- **NppStatus** **nppiWarpAffineQuad_32f_P3R** (const **Npp32f** *pSrc[3], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32f** *pDst[3], int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit float, three planes).

- **NppStatus** **nppiWarpAffineQuad_32f_P4R** (const **Npp32f** *pSrc[4], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32f** *pDst[4], int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit float, four planes).

- `NppStatus nppiWarpAffine_32s_C1R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit signed integer, single channel).

- `NppStatus nppiWarpAffine_32s_C3R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int interpolation)

Affine transform of an image (32bit signed integer, three channels).

- `NppStatus nppiWarpAffine_32s_C4R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit signed integer, four channels).

- `NppStatus nppiWarpAffine_32s_AC4R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit signed integer, four channels RGBA).

- `NppStatus nppiWarpAffine_32s_P3R` (const `Npp32s` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit signed integer, three planes).

- `NppStatus nppiWarpAffine_32s_P4R` (const `Npp32s` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Affine transform of an image (32bit signed integer, four planes).

- `NppStatus nppiWarpAffineBack_32s_C1R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit signed integer, single channel).

- `NppStatus nppiWarpAffineBack_32s_C3R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit signed integer, three channels).

- `NppStatus nppiWarpAffineBack_32s_C4R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit signed integer, four channels).

- `NppStatus nppiWarpAffineBack_32s_AC4R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit signed integer, four channels RGBA).

- `NppStatus nppiWarpAffineBack_32s_P3R` (const `Npp32s` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit signed integer, three planes).

- `NppStatus nppiWarpAffineBack_32s_P4R` (const `Npp32s` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[2][3], int eInterpolation)

Inverse affine transform of an image (32bit signed integer, four planes).

- `NppStatus nppiWarpAffineQuad_32s_C1R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit signed integer, single channel).

- `NppStatus nppiWarpAffineQuad_32s_C3R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit signed integer, three channels).

- `NppStatus nppiWarpAffineQuad_32s_C4R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit signed integer, four channels).

- `NppStatus nppiWarpAffineQuad_32s_AC4R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit signed integer, four channels RGBA).

- `NppStatus nppiWarpAffineQuad_32s_P3R` (const `Npp32s` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32s` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit signed integer, three planes).

- `NppStatus nppiWarpAffineQuad_32s_P4R` (const `Npp32s` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32s` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Affine transform of an image (32bit signed integer, four planes).

Perspective warping, perspective transform calculation

Perspective warping of an image is the transform of image pixel positions, defined by the following formulas:

$$X_{new} = \frac{C_{00} * x + C_{01} * y + C_{02}}{C_{20} * x + C_{21} * y + C_{22}} \quad Y_{new} = \frac{C_{10} * x + C_{11} * y + C_{12}}{C_{20} * x + C_{21} * y + C_{22}} \quad C = \begin{bmatrix} C_{00} & C_{01} & C_{02} \\ C_{10} & C_{11} & C_{12} \\ C_{20} & C_{21} & C_{22} \end{bmatrix}$$

That is, any pixel of the transformed image with coordinates (X_{new}, Y_{new}) has a preimage with coordinates (x, y) .

The mapping C is fully defined by 8 values $C_{ij}, (i, j) = \overline{0, 2}$, except of C_{22} , which is a normalizer. The transform has a property of mapping any convex quadrangle to a convex quadrangle, which is used in a group of functions `npplWarpPerspectiveQuad`. The NPPI implementation of perspective transform has some issues which are discussed in each function's documentation.

- **NppStatus npplGetPerspectiveTransform** (**NppiRect** srcRoi, const double quad[4][2], double coeffs[3][3])
Calculates perspective transform coefficients given source rectangular ROI and its destination quadrangle projection.
- **NppStatus npplGetPerspectiveQuad** (**NppiRect** srcRoi, double quad[4][2], const double coeffs[3][3])
Calculates perspective transform projection of given source rectangular ROI.
- **NppStatus npplGetPerspectiveBound** (**NppiRect** srcRoi, double bound[2][2], const double coeffs[3][3])
Calculates bounding box of the perspective transform projection of the given source rectangular ROI.
- **NppStatus npplWarpPerspective_8u_C1R** (const **Npp8u** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp8u** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)
Perspective transform of an image (8bit unsigned integer, single channel).
- **NppStatus npplWarpPerspective_8u_C3R** (const **Npp8u** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp8u** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)
Perspective transform of an image (8bit unsigned integer, three channels).
- **NppStatus npplWarpPerspective_8u_C4R** (const **Npp8u** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp8u** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)
Perspective transform of an image (8bit unsigned integer, four channels).
- **NppStatus npplWarpPerspective_8u_AC4R** (const **Npp8u** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp8u** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)
Perspective transform of an image (8bit unsigned integer, four channels RGBA).
- **NppStatus npplWarpPerspective_8u_P3R** (const **Npp8u** *pSrc[3], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp8u** *pDst[3], int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)
Perspective transform of an image (8bit unsigned integer, three planes).
- **NppStatus npplWarpPerspective_8u_P4R** (const **Npp8u** *pSrc[4], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp8u** *pDst[4], int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)
Perspective transform of an image (8bit unsigned integer, four planes).
- **NppStatus npplWarpPerspectiveBack_8u_C1R** (const **Npp8u** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp8u** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (8bit unsigned integer, single channel).

- `NppStatus nppiWarpPerspectiveBack_8u_C3R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (8bit unsigned integer, three channels).

- `NppStatus nppiWarpPerspectiveBack_8u_C4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (8bit unsigned integer, four channels).

- `NppStatus nppiWarpPerspectiveBack_8u_AC4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (8bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpPerspectiveBack_8u_P3R` (const `Npp8u` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (8bit unsigned integer, three planes).

- `NppStatus nppiWarpPerspectiveBack_8u_P4R` (const `Npp8u` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp8u` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (8bit unsigned integer, four planes).

- `NppStatus nppiWarpPerspectiveQuad_8u_C1R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (8bit unsigned integer, single channel).

- `NppStatus nppiWarpPerspectiveQuad_8u_C3R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (8bit unsigned integer, three channels).

- `NppStatus nppiWarpPerspectiveQuad_8u_C4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (8bit unsigned integer, four channels).

- `NppStatus nppiWarpPerspectiveQuad_8u_AC4R` (const `Npp8u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (8bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpPerspectiveQuad_8u_P3R` (const `Npp8u` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (8bit unsigned integer, three planes).

- `NppStatus nppiWarpPerspectiveQuad_8u_P4R` (const `Npp8u` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp8u` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (8bit unsigned integer, four planes).

- `NppStatus nppiWarpPerspective_16u_C1R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, single channel).

- `NppStatus nppiWarpPerspective_16u_C3R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, three channels).

- `NppStatus nppiWarpPerspective_16u_C4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, four channels).

- `NppStatus nppiWarpPerspective_16u_AC4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpPerspective_16u_P3R` (const `Npp16u` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, three planes).

- `NppStatus nppiWarpPerspective_16u_P4R` (const `Npp16u` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, four planes).

- `NppStatus nppiWarpPerspectiveBack_16u_C1R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (16bit unsigned integer, single channel).

- `NppStatus nppiWarpPerspectiveBack_16u_C3R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (16bit unsigned integer, three channels).

- `NppStatus nppiWarpPerspectiveBack_16u_C4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (16bit unsigned integer, four channels).

- `NppStatus nppiWarpPerspectiveBack_16u_AC4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (16bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpPerspectiveBack_16u_P3R` (const `Npp16u` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (16bit unsigned integer, three planes).

- `NppStatus nppiWarpPerspectiveBack_16u_P4R` (const `Npp16u` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp16u` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (16bit unsigned integer, four planes).

- `NppStatus nppiWarpPerspectiveQuad_16u_C1R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, single channel).

- `NppStatus nppiWarpPerspectiveQuad_16u_C3R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, three channels).

- `NppStatus nppiWarpPerspectiveQuad_16u_C4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, four channels).

- `NppStatus nppiWarpPerspectiveQuad_16u_AC4R` (const `Npp16u` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, four channels RGBA).

- `NppStatus nppiWarpPerspectiveQuad_16u_P3R` (const `Npp16u` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, three planes).

- `NppStatus nppiWarpPerspectiveQuad_16u_P4R` (const `Npp16u` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp16u` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (16bit unsigned integer, four planes).

- `NppStatus nppiWarpPerspective_32f_C1R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit float, single channel).

- **NppStatus** **nppiWarpPerspective_32f_C3R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit float, three channels).

- **NppStatus** **nppiWarpPerspective_32f_C4R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit float, four channels).

- **NppStatus** **nppiWarpPerspective_32f_AC4R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit float, four channels RGBA).

- **NppStatus** **nppiWarpPerspective_32f_P3R** (const **Npp32f** *pSrc[3], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst[3], int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit float, three planes).

- **NppStatus** **nppiWarpPerspective_32f_P4R** (const **Npp32f** *pSrc[4], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst[4], int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit float, four planes).

- **NppStatus** **nppiWarpPerspectiveBack_32f_C1R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit float, single channel).

- **NppStatus** **nppiWarpPerspectiveBack_32f_C3R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit float, three channels).

- **NppStatus** **nppiWarpPerspectiveBack_32f_C4R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit float, four channels).

- **NppStatus** **nppiWarpPerspectiveBack_32f_AC4R** (const **Npp32f** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst, int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit float, four channels RGBA).

- **NppStatus** **nppiWarpPerspectiveBack_32f_P3R** (const **Npp32f** *pSrc[3], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, **Npp32f** *pDst[3], int nDstStep, **NppiRect** dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit float, three planes).

- `NppStatus nppiWarpPerspectiveBack_32f_P4R` (const `Npp32f` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32f` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit float, four planes).

- `NppStatus nppiWarpPerspectiveQuad_32f_C1R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit float, single channel).

- `NppStatus nppiWarpPerspectiveQuad_32f_C3R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit float, three channels).

- `NppStatus nppiWarpPerspectiveQuad_32f_C4R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit float, four channels).

- `NppStatus nppiWarpPerspectiveQuad_32f_AC4R` (const `Npp32f` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32f` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit float, four channels RGBA).

- `NppStatus nppiWarpPerspectiveQuad_32f_P3R` (const `Npp32f` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32f` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit float, three planes).

- `NppStatus nppiWarpPerspectiveQuad_32f_P4R` (const `Npp32f` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32f` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit float, four planes).

- `NppStatus nppiWarpPerspective_32s_C1R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit signed integer, single channel).

- `NppStatus nppiWarpPerspective_32s_C3R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit signed integer, three channels).

- `NppStatus nppiWarpPerspective_32s_C4R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit signed integer, four channels).

- `NppStatus nppiWarpPerspective_32s_AC4R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit signed integer, four channels RGBA).

- `NppStatus nppiWarpPerspective_32s_P3R` (const `Npp32s` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit signed integer, three planes).

- `NppStatus nppiWarpPerspective_32s_P4R` (const `Npp32s` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Perspective transform of an image (32bit signed integer, four planes).

- `NppStatus nppiWarpPerspectiveBack_32s_C1R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit signed integer, single channel).

- `NppStatus nppiWarpPerspectiveBack_32s_C3R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit signed integer, three channels).

- `NppStatus nppiWarpPerspectiveBack_32s_C4R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit signed integer, four channels).

- `NppStatus nppiWarpPerspectiveBack_32s_AC4R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit signed integer, four channels RGBA).

- `NppStatus nppiWarpPerspectiveBack_32s_P3R` (const `Npp32s` *pSrc[3], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst[3], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit signed integer, three planes).

- `NppStatus nppiWarpPerspectiveBack_32s_P4R` (const `Npp32s` *pSrc[4], `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, `Npp32s` *pDst[4], int nDstStep, `NppiRect` dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit signed integer, four planes).

- `NppStatus nppiWarpPerspectiveQuad_32s_C1R` (const `Npp32s` *pSrc, `NppiSize` oSrcSize, int nSrcStep, `NppiRect` srcRoi, const double srcQuad[4][2], `Npp32s` *pDst, int nDstStep, `NppiRect` dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit signed integer, single channel).

- **NppStatus nppiWarpPerspectiveQuad_32s_C3R** (const **Npp32s** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32s** *pDst, int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit signed integer, three channels).

- **NppStatus nppiWarpPerspectiveQuad_32s_C4R** (const **Npp32s** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32s** *pDst, int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit signed integer, four channels).

- **NppStatus nppiWarpPerspectiveQuad_32s_AC4R** (const **Npp32s** *pSrc, **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32s** *pDst, int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit signed integer, four channels RGBA).

- **NppStatus nppiWarpPerspectiveQuad_32s_P3R** (const **Npp32s** *pSrc[3], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32s** *pDst[3], int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit signed integer, three planes).

- **NppStatus nppiWarpPerspectiveQuad_32s_P4R** (const **Npp32s** *pSrc[4], **NppiSize** oSrcSize, int nSrcStep, **NppiRect** srcRoi, const double srcQuad[4][2], **Npp32s** *pDst[4], int nDstStep, **NppiRect** dstRoi, const double dstQuad[4][2], int eInterpolation)

Perspective transform of an image (32bit signed integer, four planes).

7.14.1 Detailed Description

Routines manipulating an image's geometry.

7.14.2 Function Documentation

7.14.2.1 **NppStatus nppiGetAffineBound** (**NppiRect** srcRoi, double bound[2][2], const double coeffs[2][3])

Calculates bounding box of the affine transform projection of the given source rectangular ROI.

Parameters:

srcRoi Source ROI

bound Bounding box of the transformed source ROI

coeffs Affine transform coefficients

Returns:

Error codes:

- **NPP_SIZE_ERROR** Indicates an error condition if any image dimension has zero or negative value
- **NPP_RECT_ERROR** Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- **NPP_COEFF_ERROR** Indicates an error condition if coefficient values are invalid

7.14.2.2 NppStatus nppiGetAffineQuad (NppiRect *srcRoi*, double *quad*[4][2], const double *coeffs*[2][3])

Calculates affine transform projection of given source rectangular ROI.

Parameters:

srcRoi Source ROI
quad Destination quadrangle
coeffs Affine transform coefficients

Returns:

Error codes:

- [NPP_SIZE_ERROR](#) Indicates an error condition if any image dimension has zero or negative value
- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the *srcRoi* and source image is less than or equal to 1
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid

7.14.2.3 NppStatus nppiGetAffineTransform (NppiRect *srcRoi*, const double *quad*[4][2], double *coeffs*[2][3])

Calculates affine transform coefficients given source rectangular ROI and its destination quadrangle projection.

Parameters:

srcRoi Source ROI
quad Destination quadrangle
coeffs Affine transform coefficients

Returns:

Error codes:

- [NPP_SIZE_ERROR](#) Indicates an error condition if any image dimension has zero or negative value
- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the *srcRoi* and source image is less than or equal to 1
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_AFFINE_QUAD_INCORRECT_WARNING](#) Indicates a warning when *quad* does not conform to the transform properties. Fourth vertex is ignored, internally computed coordinates are used instead

7.14.2.4 NppStatus nppiGetPerspectiveBound (NppiRect *srcRoi*, double *bound*[2][2], const double *coeffs*[3][3])

Calculates bounding box of the perspective transform projection of the given source rectangular ROI.

Parameters:

srcRoi Source ROI
bound Bounding box of the transformed source ROI
coeffs Perspective transform coefficients

Returns:

Error codes:

- [NPP_SIZE_ERROR](#) Indicates an error condition if any image dimension has zero or negative value
- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the *srcRoi* and source image is less than or equal to 1
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid

7.14.2.5 NppStatus nppiGetPerspectiveQuad (NppiRect *srcRoi*, double *quad*[4][2], const double *coeffs*[3][3])

Calculates perspective transform projection of given source rectangular ROI.

Parameters:

srcRoi Source ROI
quad Destination quadrangle
coeffs Perspective transform coefficients

Returns:

Error codes:

- [NPP_SIZE_ERROR](#) Indicates an error condition if any image dimension has zero or negative value
- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the *srcRoi* and source image is less than or equal to 1
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid

7.14.2.6 NppStatus nppiGetPerspectiveTransform (NppiRect *srcRoi*, const double *quad*[4][2], double *coeffs*[3][3])

Calculates perspective transform coefficients given source rectangular ROI and its destination quadrangle projection.

Parameters:

srcRoi Source ROI
quad Destination quadrangle
coeffs Perspective transform coefficients

Returns:

Error codes:

- [NPP_SIZE_ERROR](#) Indicates an error condition if any image dimension has zero or negative value
- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid

7.14.2.7 NppStatus nppiGetRotateBound (NppiRect oSrcROI, double aBoundingBox[2][2], double nAngle, double nShiftX, double nShiftY)

Compute bounding-box of rotated image.

Parameters:

oSrcROI Region-of-interest of the source image.

aBoundingBox Two 2D points representing the bounding-box of the rotated image. All four points from nppiGetRotateQuad are contained inside the axis-aligned rectangle spanned by the two points of this bounding box.

nAngle The rotation angle.

nShiftX Post-rotation shift in x-direction.

nShiftY Post-rotation shift in y-direction.

Returns:

[ROI Related Error Codes.](#)

7.14.2.8 NppStatus nppiGetRotateQuad (NppiRect oSrcROI, double aQuad[4][2], double nAngle, double nShiftX, double nShiftY)

Compute shape of rotated image.

Parameters:

oSrcROI Region-of-interest of the source image.

aQuad Array of 2D points. These points are the locations of the corners of the rotated ROI.

nAngle The rotation nAngle.

nShiftX Post-rotation shift in x-direction

nShiftY Post-rotation shift in y-direction

Returns:

[ROI Related Error Codes.](#)

7.14.2.9 NppStatus nppiMirror_16u_AC4R (const Npp16u *pSrc, int nSrcStep, Npp16u *pDst, int nDstStep, NppiSize oROI, NppiAxis flip)

4 channel 16-bit unsigned image mirror not affecting alpha.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Distance in bytes between starts of consecutive lines of the destination image.
oROI Region-of-Interest (ROI).
flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- `NPP_MIRROR_FLIP_ERR` if flip has an illegal value.

7.14.2.10 `NppStatus nppiMirror_16u_C1R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oROI, NppiAxis flip)`

16-bit unsigned image mirror.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oROI Region-of-Interest (ROI).
flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- `NPP_MIRROR_FLIP_ERR` if flip has an illegal value.

7.14.2.11 `NppStatus nppiMirror_16u_C3R (const Npp16u * pSrc, int nSrcStep, Npp16u * pDst, int nDstStep, NppiSize oROI, NppiAxis flip)`

3 channel 16-bit unsigned image mirror.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oROI Region-of-Interest (ROI).
flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- `NPP_MIRROR_FLIP_ERR` if flip has an illegal value.

7.14.2.12 NppStatus nppiMirror_16u_C4R (const Npp16u * *pSrc*, int *nSrcStep*, Npp16u * *pDst*, int *nDstStep*, NppiSize *oROI*, NppiAxis *flip*)

4 channel 16-bit unsigned image mirror.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Distance in bytes between starts of consecutive lines of the destination image.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- **NPP_MIRROR_FLIP_ERR** if flip has an illegal value.

7.14.2.13 NppStatus nppiMirror_32f_AC4R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oROI*, NppiAxis *flip*)

4 channel 32-bit float image mirror not affecting alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Distance in bytes between starts of consecutive lines of the destination image.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- **NPP_MIRROR_FLIP_ERR** if flip has an illegal value.

7.14.2.14 NppStatus nppiMirror_32f_C1R (const Npp32f * *pSrc*, int *nSrcStep*, Npp32f * *pDst*, int *nDstStep*, NppiSize *oROI*, NppiAxis *flip*)

32-bit float image mirror.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- `NPP_MIRROR_FLIP_ERR` if flip has an illegal value.

7.14.2.15 NppStatus nppiMirror_32f_C3R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oROI, NppiAxis flip)

3 channel 32-bit float image mirror.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- `NPP_MIRROR_FLIP_ERR` if flip has an illegal value.

7.14.2.16 NppStatus nppiMirror_32f_C4R (const Npp32f * pSrc, int nSrcStep, Npp32f * pDst, int nDstStep, NppiSize oROI, NppiAxis flip)

4 channel 32-bit float image mirror.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Distance in bytes between starts of consecutive lines of the destination image.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- `NPP_MIRROR_FLIP_ERR` if flip has an illegal value.

7.14.2.17 NppStatus nppiMirror_32s_AC4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oROI, NppiAxis flip)

4 channel 32-bit image mirror not affecting alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Distance in bytes between starts of consecutive lines of the destination image.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- **NPP_MIRROR_FLIP_ERR** if flip has an illegal value.

7.14.2.18 NppStatus nppiMirror_32s_C1R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oROI, NppiAxis flip)

32-bit image mirror.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- **NPP_MIRROR_FLIP_ERR** if flip has an illegal value.

7.14.2.19 NppStatus nppiMirror_32s_C3R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oROI, NppiAxis flip)

3 channel 32-bit image mirror.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- `NPP_MIRROR_FLIP_ERR` if flip has an illegal value.

7.14.2.20 NppStatus nppiMirror_32s_C4R (const Npp32s * pSrc, int nSrcStep, Npp32s * pDst, int nDstStep, NppiSize oROI, NppiAxis flip)

4 channel 32-bit image mirror.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Distance in bytes between starts of consecutive lines of the destination image.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- `NPP_MIRROR_FLIP_ERR` if flip has an illegal value.

7.14.2.21 NppStatus nppiMirror_8u_AC4R (const Npp8u * pSrc, int nSrcStep, Npp8u * pDst, int nDstStep, NppiSize oROI, NppiAxis flip)

4 channel 8-bit unsigned image mirror not affecting alpha.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Distance in bytes between starts of consecutive lines of the destination image.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- `NPP_MIRROR_FLIP_ERR` if flip has an illegal value.

7.14.2.22 NppStatus nppiMirror_8u_C1R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oROI*, NppiAxis *flip*)

8-bit unsigned image mirror.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- **NPP_MIRROR_FLIP_ERR** if flip has an illegal value.

7.14.2.23 NppStatus nppiMirror_8u_C3R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oROI*, NppiAxis *flip*)

3 channel 8-bit unsigned image mirror.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oROI Region-of-Interest (ROI).

flip Specifies the axis about which the image is to be mirrored.

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- **NPP_MIRROR_FLIP_ERR** if flip has an illegal value.

7.14.2.24 NppStatus nppiMirror_8u_C4R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oROI*, NppiAxis *flip*)

4 channel 8-bit unsigned image mirror.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Distance in bytes between starts of consecutive lines of the destination image.

oROI [Region-of-Interest \(ROI\)](#).

flip Specifies the axis about which the image is to be mirrored.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_MIRROR_FLIP_ERR](#) if flip has an illegal value.

7.14.2.25 `NppStatus nppiResize_8u_C1R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect oSrcROI, Npp8u * pDst, int nDstStep, NppiSize dstROISize, double xFactor, double yFactor, int eInterpolation)`

8-bit unsigned image resize.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSize Size in pixels of the source image

oSrcROI Region of interest in the source image.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstROISize Size in pixels of the destination image

xFactor Factors by which x dimension is changed

yFactor Factors by which y dimension is changed

eInterpolation The type of eInterpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) indicates an error condition if srcROIRect has no intersection with the source image.
- [NPP_RESIZE_NO_OPERATION_ERROR](#) if either destination ROI width or height is less than 1 pixel.
- [NPP_RESIZE_FACTOR_ERROR](#) Indicates an error condition if either xFactor or yFactor is less than or equal to zero.
- [NPP_INTERPOLATION_ERROR](#) if eInterpolation has an illegal value.

7.14.2.26 `NppStatus nppiResize_8u_C4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect oSrcROI, Npp8u * pDst, int nDstStep, NppiSize dstROISize, double xFactor, double yFactor, int eInterpolation)`

4 channel 8-bit unsigned image resize.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSize Size in pixels of the source image

oSrcROI Region of interest in the source image.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstROISize Size in pixels of the destination image

xFactor Factors by which x dimension is changed

yFactor Factors by which y dimension is changed

eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) indicates an error condition if srcROIRect has no intersection with the source image.
- [NPP_RESIZE_NO_OPERATION_ERROR](#) if either destination ROI width or height is less than 1 pixel.
- [NPP_RESIZE_FACTOR_ERROR](#) Indicates an error condition if either xFactor or yFactor is less than or equal to zero.
- [NPP_INTERPOLATION_ERROR](#) if eInterpolation has an illegal value.

7.14.2.27 `NppStatus nppiRotate_16u_AC4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect oSrcROI, Npp16u * pDst, int nDstStep, NppiRect oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)`

4 channel 16-bit unsigned image rotate ignoring alpha channel.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSize Size in pixels of the source image

oSrcROI Region of interest in the source image.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oDstROI Region of interest in the destination image.

nAngle The angle of rotation in degrees.

nShiftX Shift along horizontal axis

nShiftY Shift along vertical axis

eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.28 `NppStatus nppiRotate_16u_C1R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect oSrcROI, Npp16u * pDst, int nDstStep, NppiRect oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)`

16-bit unsigned image rotate.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSize Size in pixels of the source image
oSrcROI Region of interest in the source image.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oDstROI Region of interest in the destination image.
nAngle The angle of rotation in degrees.
nShiftX Shift along horizontal axis
nShiftY Shift along vertical axis
eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.29 `NppStatus nppiRotate_16u_C3R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect oSrcROI, Npp16u * pDst, int nDstStep, NppiRect oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)`

3 channel 16-bit unsigned image rotate.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSize Size in pixels of the source image
oSrcROI Region of interest in the source image.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oDstROI Region of interest in the destination image.
nAngle The angle of rotation in degrees.
nShiftX Shift along horizontal axis
nShiftY Shift along vertical axis
eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.30 `NppStatus nppiRotate_16u_C4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect oSrcROI, Npp16u * pDst, int nDstStep, NppiRect oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)`

4 channel 16-bit unsigned image rotate.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSize Size in pixels of the source image

oSrcROI Region of interest in the source image.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstROI Region of interest in the destination image.

nAngle The angle of rotation in degrees.

nShiftX Shift along horizontal axis

nShiftY Shift along vertical axis

eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.31 `NppStatus nppiRotate_32f_AC4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect oSrcROI, Npp32f * pDst, int nDstStep, NppiRect oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)`

4 channel 32-bit float image rotate ignoring alpha channel.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

oSrcSize Size in pixels of the source image

oSrcROI Region of interest in the source image.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oDstROI Region of interest in the destination image.

nAngle The angle of rotation in degrees.

nShiftX Shift along horizontal axis

nShiftY Shift along vertical axis

eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.32 NppStatus nppiRotate_32f_C1R (const Npp32f * *pSrc*, NppiSize *oSrcSize*, int *nSrcStep*, NppiRect *oSrcROI*, Npp32f * *pDst*, int *nDstStep*, NppiRect *oDstROI*, double *nAngle*, double *nShiftX*, double *nShiftY*, int *eInterpolation*)

32-bit float image rotate.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSize Size in pixels of the source image
oSrcROI Region of interest in the source image.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oDstROI Region of interest in the destination image.
nAngle The angle of rotation in degrees.
nShiftX Shift along horizontal axis
nShiftY Shift along vertical axis
eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.33 NppStatus nppiRotate_32f_C3R (const Npp32f * *pSrc*, NppiSize *oSrcSize*, int *nSrcStep*, NppiRect *oSrcROI*, Npp32f * *pDst*, int *nDstStep*, NppiRect *oDstROI*, double *nAngle*, double *nShiftX*, double *nShiftY*, int *eInterpolation*)

3 channel 32-bit float image rotate.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSize Size in pixels of the source image
oSrcROI Region of interest in the source image.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oDstROI Region of interest in the destination image.
nAngle The angle of rotation in degrees.
nShiftX Shift along horizontal axis
nShiftY Shift along vertical axis
eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.34 NppStatus nppiRotate_32f_C4R (const Npp32f * *pSrc*, NppiSize *oSrcSize*, int *nSrcStep*, NppiRect *oSrcROI*, Npp32f * *pDst*, int *nDstStep*, NppiRect *oDstROI*, double *nAngle*, double *nShiftX*, double *nShiftY*, int *eInterpolation*)

4 channel 32-bit float image rotate.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSize Size in pixels of the source image
oSrcROI Region of interest in the source image.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oDstROI Region of interest in the destination image.
nAngle The angle of rotation in degrees.
nShiftX Shift along horizontal axis
nShiftY Shift along vertical axis
eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.35 NppStatus nppiRotate_8u_AC4R (const Npp8u * *pSrc*, NppiSize *oSrcSize*, int *nSrcStep*, NppiRect *oSrcROI*, Npp8u * *pDst*, int *nDstStep*, NppiRect *oDstROI*, double *nAngle*, double *nShiftX*, double *nShiftY*, int *eInterpolation*)

4 channel 8-bit unsigned image rotate ignoring alpha channel.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSize Size in pixels of the source image
oSrcROI Region of interest in the source image.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oDstROI Region of interest in the destination image.
nAngle The angle of rotation in degrees.
nShiftX Shift along horizontal axis
nShiftY Shift along vertical axis
eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.36 `NppStatus nppiRotate_8u_C1R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect oSrcROI, Npp8u * pDst, int nDstStep, NppiRect oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)`

8-bit unsigned image rotate.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSize Size in pixels of the source image
oSrcROI Region of interest in the source image.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oDstROI Region of interest in the destination image.
nAngle The angle of rotation in degrees.
nShiftX Shift along horizontal axis
nShiftY Shift along vertical axis
eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.14.2.37 `NppStatus nppiRotate_8u_C3R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect oSrcROI, Npp8u * pDst, int nDstStep, NppiRect oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)`

3 channel 8-bit unsigned image rotate.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
oSrcSize Size in pixels of the source image
oSrcROI Region of interest in the source image.
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oDstROI Region of interest in the destination image.
nAngle The angle of rotation in degrees.
nShiftX Shift along horizontal axis
nShiftY Shift along vertical axis
eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.38 `NppStatus nppiRotate_8u_C4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect oSrcROI, Npp8u * pDst, int nDstStep, NppiRect oDstROI, double nAngle, double nShiftX, double nShiftY, int eInterpolation)`

4 channel 8-bit unsigned image rotate.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

oSrcSize Size in pixels of the source image

oSrcROI Region of interest in the source image.

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oDstROI Region of interest in the destination image.

nAngle The angle of rotation in degrees.

nShiftX Shift along horizontal axis

nShiftY Shift along vertical axis

eInterpolation The type of interpolation to perform resampling

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#), [Rotate specific Error Codes](#)

7.14.2.39 `NppStatus nppiWarpAffine_16u_AC4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, four channels RGBA).

See also:

[nppiWarpAffine_16u_C1R](#)

7.14.2.40 `NppStatus nppiWarpAffine_16u_C1R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetAffineTransform` function or set explicitly. The function operates on source and destination regions of interest. The affine warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$X_{new} = C_{00} * x + C_{01} * y + C_{02} \quad Y_{new} = C_{10} * x + C_{11} * y + C_{12}$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetAffineQuad` and `nppiGetAffineBound` can help with destination ROI specification.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but does not perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void*)(pDst + dstRoi.x))` and `(int)((void*)(pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Affine transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the *srcRoi* and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if *srcRoi* has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if *eInterpolation* has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.41 `NppStatus nppiWarpAffine_16u_C3R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, three channels).

See also:

[nppiWarpAffine_16u_C1R](#)

7.14.2.42 `NppStatus nppiWarpAffine_16u_C4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, four channels).

See also:

[nppiWarpAffine_16u_C1R](#)

7.14.2.43 `NppStatus nppiWarpAffine_16u_P3R (const Npp16u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, three planes).

See also:

[nppiWarpAffine_16u_C1R](#)

7.14.2.44 `NppStatus nppiWarpAffine_16u_P4R (const Npp16u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, four planes).

See also:

[nppiWarpAffine_16u_C1R](#)

7.14.2.45 `NppStatus nppiWarpAffine_32f_AC4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (32bit float, four channels RGBA).

See also:

[nppiWarpAffine_32f_C1R](#)

7.14.2.46 `NppStatus nppiWarpAffine_32f_C1R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (32bit float, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetAffineTransform` function or set explicitly. The function operates on source and destination regions of interest.

The affine warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$X_{new} = C_{00} * x + C_{01} * y + C_{02} \quad Y_{new} = C_{10} * x + C_{11} * y + C_{12}$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetAffineQuad` and `nppiGetAffineBound` can help with destination ROI specification.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Affine transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the `srcRoi` and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if `srcRoi` has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if `eInterpolation` has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.47 `NppStatus nppiWarpAffine_32f_C3R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (32bit float, three channels).

See also:

[nppiWarpAffine_32f_C1R](#)

7.14.2.48 `NppStatus nppiWarpAffine_32f_C4R` (`const Npp32f * pSrc`, `NppiSize oSrcSize`, `int nSrcStep`, `NppiRect srcRoi`, `Npp32f * pDst`, `int nDstStep`, `NppiRect dstRoi`, `const double coeffs[2][3]`, `int eInterpolation`)

Affine transform of an image (32bit float, four channels).

See also:

[nppiWarpAffine_32f_C1R](#)

7.14.2.49 `NppStatus nppiWarpAffine_32f_P3R` (`const Npp32f * pSrc[3]`, `NppiSize oSrcSize`, `int nSrcStep`, `NppiRect srcRoi`, `Npp32f * pDst[3]`, `int nDstStep`, `NppiRect dstRoi`, `const double coeffs[2][3]`, `int eInterpolation`)

Affine transform of an image (32bit float, three planes).

See also:

[nppiWarpAffine_32f_C1R](#)

7.14.2.50 `NppStatus nppiWarpAffine_32f_P4R` (`const Npp32f * pSrc[4]`, `NppiSize oSrcSize`, `int nSrcStep`, `NppiRect srcRoi`, `Npp32f * pDst[4]`, `int nDstStep`, `NppiRect dstRoi`, `const double coeffs[2][3]`, `int eInterpolation`)

Affine transform of an image (32bit float, four planes).

See also:

[nppiWarpAffine_32f_C1R](#)

7.14.2.51 `NppStatus nppiWarpAffine_32s_AC4R` (`const Npp32s * pSrc`, `NppiSize oSrcSize`, `int nSrcStep`, `NppiRect srcRoi`, `Npp32s * pDst`, `int nDstStep`, `NppiRect dstRoi`, `const double coeffs[2][3]`, `int eInterpolation`)

Affine transform of an image (32bit signed integer, four channels RGBA).

See also:

[nppiWarpAffine_32s_C1R](#)

7.14.2.52 `NppStatus nppiWarpAffine_32s_C1R` (`const Npp32s * pSrc`, `NppiSize oSrcSize`, `int nSrcStep`, `NppiRect srcRoi`, `Npp32s * pDst`, `int nDstStep`, `NppiRect dstRoi`, `const double coeffs[2][3]`, `int eInterpolation`)

Affine transform of an image (32bit signed integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetAffineTransform` function or set explicitly. The function operates on source and destination regions of interest.

The affine warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$X_{new} = C_{00} * x + C_{01} * y + C_{02} \quad Y_{new} = C_{10} * x + C_{11} * y + C_{12}$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetAffineQuad` and `nppiGetAffineBound` can help with destination ROI specification.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Affine transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the `srcRoi` and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if `srcRoi` has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if `eInterpolation` has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.53 `NppStatus nppiWarpAffine_32s_C3R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int interpolation)`

Affine transform of an image (32bit signed integer, three channels).

See also:

[nppiWarpAffine_32s_C1R](#)

7.14.2.54 `NppStatus nppiWarpAffine_32s_C4R` (`const Npp32s * pSrc`, `NppiSize oSrcSize`, `int nSrcStep`, `NppiRect srcRoi`, `Npp32s * pDst`, `int nDstStep`, `NppiRect dstRoi`, `const double coeffs[2][3]`, `int eInterpolation`)

Affine transform of an image (32bit signed integer, four channels).

See also:

[nppiWarpAffine_32s_C1R](#)

7.14.2.55 `NppStatus nppiWarpAffine_32s_P3R` (`const Npp32s * pSrc[3]`, `NppiSize oSrcSize`, `int nSrcStep`, `NppiRect srcRoi`, `Npp32s * pDst[3]`, `int nDstStep`, `NppiRect dstRoi`, `const double coeffs[2][3]`, `int eInterpolation`)

Affine transform of an image (32bit signed integer, three planes).

See also:

[nppiWarpAffine_32s_C1R](#)

7.14.2.56 `NppStatus nppiWarpAffine_32s_P4R` (`const Npp32s * pSrc[4]`, `NppiSize oSrcSize`, `int nSrcStep`, `NppiRect srcRoi`, `Npp32s * pDst[4]`, `int nDstStep`, `NppiRect dstRoi`, `const double coeffs[2][3]`, `int eInterpolation`)

Affine transform of an image (32bit signed integer, four planes).

See also:

[nppiWarpAffine_32s_C1R](#)

7.14.2.57 `NppStatus nppiWarpAffine_8u_AC4R` (`const Npp8u * pSrc`, `NppiSize oSrcSize`, `int nSrcStep`, `NppiRect srcRoi`, `Npp8u * pDst`, `int nDstStep`, `NppiRect dstRoi`, `const double coeffs[2][3]`, `int eInterpolation`)

Affine transform of an image (8bit unsigned integer, four channels RGBA).

See also:

[nppiWarpAffine_8u_C1R](#)

7.14.2.58 `NppStatus nppiWarpAffine_8u_C1R` (`const Npp8u * pSrc`, `NppiSize oSrcSize`, `int nSrcStep`, `NppiRect srcRoi`, `Npp8u * pDst`, `int nDstStep`, `NppiRect dstRoi`, `const double coeffs[2][3]`, `int eInterpolation`)

Affine transform of an image (8bit unsigned integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetAffineTransform` function or set explicitly. The function operates on source and destination regions of interest.

The affine warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$X_{new} = C_{00} * x + C_{01} * y + C_{02} \quad Y_{new} = C_{10} * x + C_{11} * y + C_{12}$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetAffineQuad` and `nppiGetAffineBound` can help with destination ROI specification.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but does not perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void*)(pDst + dstRoi.x))` and `(int)((void*)(pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Affine transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the `srcRoi` and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if `srcRoi` has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if interpolation has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.59 `NppStatus nppiWarpAffine_8u_C3R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (8bit unsigned integer, three channels).

See also:

[nppiWarpAffine_8u_C1R](#)

7.14.2.60 `NppStatus nppiWarpAffine_8u_C4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (8bit unsigned integer, four channels).

See also:

[nppiWarpAffine_8u_C1R](#)

7.14.2.61 `NppStatus nppiWarpAffine_8u_P3R (const Npp8u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (8bit unsigned integer, three planes).

See also:

[nppiWarpAffine_8u_C1R](#)

7.14.2.62 `NppStatus nppiWarpAffine_8u_P4R (const Npp8u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Affine transform of an image (8bit unsigned integer, four planes).

See also:

[nppiWarpAffine_8u_C1R](#)

7.14.2.63 `NppStatus nppiWarpAffineBack_16u_AC4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (16bit unsigned integer, four channels RGBA).

See also:

[nppiWarpAffineBack_16u_C1R](#)

7.14.2.64 **NppStatus nppiWarpAffineBack_16u_C1R** (const Npp16u * *pSrc*, NppiSize *oSrcSize*, int *nSrcStep*, NppiRect *srcRoi*, Npp16u * *pDst*, int *nDstStep*, NppiRect *dstRoi*, const double *coeffs*[2][3], int *eInterpolation*)

Inverse affine transform of an image (16bit unsigned integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetAffineTransform` function or set explicitly. Thus there is no need to invert coefficients in your application before calling `WarpAffineBack`. The function operates on source and destination regions of interest. The affine warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$C_{00} * X_{new} + C_{01} * Y_{new} + C_{02} = x \quad C_{10} * X_{new} + C_{11} * Y_{new} + C_{12} = y$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetAffineQuad` and `nppiGetAffineBound` can help with destination ROI specification.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but doesn't perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void*)(pDst + dstRoi.x))` and `(int)((void*)(pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

- pSrc* [Source-Image Pointer](#).
- oSrcSize* Size of source image in pixels
- nSrcStep* [Source-Image Line Step](#).
- srcRoi* Source ROI
- pDst* [Destination-Image Pointer](#).
- nDstStep* [Destination-Image Line Step](#).
- dstRoi* Destination ROI
- coeffs* Affine transform coefficients
- eInterpolation* Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the *srcRoi* and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if *srcRoi* has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if *eInterpolation* has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid

- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.65 `NppStatus nppiWarpAffineBack_16u_C3R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (16bit unsigned integer, three channels).

See also:

[nppiWarpAffineBack_16u_C1R](#)

7.14.2.66 `NppStatus nppiWarpAffineBack_16u_C4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (16bit unsigned integer, four channels).

See also:

[nppiWarpAffineBack_16u_C1R](#)

7.14.2.67 `NppStatus nppiWarpAffineBack_16u_P3R (const Npp16u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (16bit unsigned integer, three planes).

See also:

[nppiWarpAffineBack_16u_C1R](#)

7.14.2.68 `NppStatus nppiWarpAffineBack_16u_P4R (const Npp16u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (16bit unsigned integer, four planes).

See also:

[nppiWarpAffineBack_16u_C1R](#)

7.14.2.69 `NppStatus nppiWarpAffineBack_32f_AC4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit float, four channels RGBA).

See also:

[nppiWarpAffineBack_32f_C1R](#)

7.14.2.70 `NppStatus nppiWarpAffineBack_32f_C1R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit float, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetAffineTransform` function or set explicitly. Thus there is no need to invert coefficients in your application before calling `WarpAffineBack`. The function operates on source and destination regions of interest. The affine warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$C_{00} * X_{new} + C_{01} * Y_{new} + C_{02} = x \quad C_{10} * X_{new} + C_{11} * Y_{new} + C_{12} = y$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetAffineQuad` and `nppiGetAffineBound` can help with destination ROI specification.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Affine transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the `srcRoi` and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if `srcRoi` has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if `eInterpolation` has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid

- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.71 `NppStatus nppiWarpAffineBack_32f_C3R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit float, three channels).

See also:

[nppiWarpAffineBack_32f_C1R](#)

7.14.2.72 `NppStatus nppiWarpAffineBack_32f_C4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit float, four channels).

See also:

[nppiWarpAffineBack_32f_C1R](#)

7.14.2.73 `NppStatus nppiWarpAffineBack_32f_P3R (const Npp32f * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit float, three planes).

See also:

[nppiWarpAffineBack_32f_C1R](#)

7.14.2.74 `NppStatus nppiWarpAffineBack_32f_P4R (const Npp32f * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit float, four planes).

See also:

[nppiWarpAffineBack_32f_C1R](#)

7.14.2.75 `NppStatus nppiWarpAffineBack_32s_AC4R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit signed integer, four channels RGBA).

See also:

[nppiWarpAffineBack_32s_C1R](#)

7.14.2.76 `NppStatus nppiWarpAffineBack_32s_C1R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit signed integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetAffineTransform` function or set explicitly. Thus there is no need to invert coefficients in your application before calling `WarpAffineBack`. The function operates on source and destination regions of interest. The affine warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$C_{00} * X_{new} + C_{01} * Y_{new} + C_{02} = x \quad C_{10} * X_{new} + C_{11} * Y_{new} + C_{12} = y$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetAffineQuad` and `nppiGetAffineBound` can help with destination ROI specification.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Affine transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the `srcRoi` and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if `srcRoi` has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if `eInterpolation` has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid

- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.77 `NppStatus nppiWarpAffineBack_32s_C3R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit signed integer, three channels).

See also:

[nppiWarpAffineBack_32s_C1R](#)

7.14.2.78 `NppStatus nppiWarpAffineBack_32s_C4R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit signed integer, four channels).

See also:

[nppiWarpAffineBack_32s_C1R](#)

7.14.2.79 `NppStatus nppiWarpAffineBack_32s_P3R (const Npp32s * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit signed integer, three planes).

See also:

[nppiWarpAffineBack_32s_C1R](#)

7.14.2.80 `NppStatus nppiWarpAffineBack_32s_P4R (const Npp32s * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (32bit signed integer, four planes).

See also:

[nppiWarpAffineBack_32s_C1R](#)

7.14.2.81 `NppStatus nppiWarpAffineBack_8u_AC4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (8bit unsigned integer, four channels RGBA).

See also:

[nppiWarpAffineBack_8u_C1R](#)

7.14.2.82 `NppStatus nppiWarpAffineBack_8u_C1R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (8bit unsigned integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetAffineTransform` function or set explicitly. Thus there is no need to invert coefficients in your application before calling `WarpAffineBack`. The function operates on source and destination regions of interest. The affine warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$C_{00} * X_{new} + C_{01} * Y_{new} + C_{02} = x \quad C_{10} * X_{new} + C_{11} * Y_{new} + C_{12} = y$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetAffineQuad` and `nppiGetAffineBound` can help with destination ROI specification.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but doesn't perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void*)(pDst + dstRoi.x))` and `(int)((void*)(pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Affine transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if srcRoi has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if interpolation has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.83 `NppStatus nppiWarpAffineBack_8u_C3R (const Npp8u *pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u *pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (8bit unsigned integer, three channels).

See also:

[nppiWarpAffineBack_8u_C1R](#)

7.14.2.84 `NppStatus nppiWarpAffineBack_8u_C4R (const Npp8u *pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u *pDst, int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (8bit unsigned integer, four channels).

See also:

[nppiWarpAffineBack_8u_C1R](#)

7.14.2.85 `NppStatus nppiWarpAffineBack_8u_P3R (const Npp8u *pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u *pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (8bit unsigned integer, three planes).

See also:

[nppiWarpAffineBack_8u_C1R](#)

7.14.2.86 `NppStatus nppiWarpAffineBack_8u_P4R (const Npp8u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[2][3], int eInterpolation)`

Inverse affine transform of an image (8bit unsigned integer, four planes).

See also:

[nppiWarpAffineBack_8u_C1R](#)

7.14.2.87 `NppStatus nppiWarpAffineQuad_16u_AC4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, four channels RGBA).

See also:

[nppiWarpAffineQuad_16u_C1R](#)

7.14.2.88 `NppStatus nppiWarpAffineQuad_16u_C1R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, single channel).

This function performs affine warping of a the specified quadrangle in the source image to the specified quadrangle in the destination image. The function `nppiWarpAffineQuad` uses the same formulas for pixel mapping as in `nppiWarpAffine` function. The transform coefficients are computed internally. The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but doesn't perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void *) (pDst + dstRoi.x))` and `(int)((void *) (pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc [Source-Image Pointer](#).
oSrcSize Size of source image in pixels
nSrcStep [Source-Image Line Step](#).
srcRoi Source ROI
srcQuad Source quadrangle
pDst [Destination-Image Pointer](#).

nDstStep Destination-Image Line Step.

dstRoi Destination ROI

dstQuad Destination quadrangle

eInterpolation Interpolation mode: can be NPPI_INTER_NN, NPPI_INTER_LINEAR or NPPI_INTER_CUBIC

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if srcRoi has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if eInterpolation has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.89 `NppStatus nppiWarpAffineQuad_16u_C3R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, three channels).

See also:

[npippiWarpAffineQuad_16u_C1R](#)

7.14.2.90 `NppStatus nppiWarpAffineQuad_16u_C4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, four channels).

See also:

[npippiWarpAffineQuad_16u_C1R](#)

7.14.2.91 `NppStatus nppiWarpAffineQuad_16u_P3R (const Npp16u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst[3], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, three planes).

See also:

[npippiWarpAffineQuad_16u_C1R](#)

7.14.2.92 `NppStatus nppiWarpAffineQuad_16u_P4R (const Npp16u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst[4], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (16bit unsigned integer, four planes).

See also:

[nppiWarpAffineQuad_16u_C1R](#)

7.14.2.93 `NppStatus nppiWarpAffineQuad_32f_AC4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit float, four channels RGBA).

See also:

[nppiWarpAffineQuad_32f_C1R](#)

7.14.2.94 `NppStatus nppiWarpAffineQuad_32f_C1R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit float, single channel).

This function performs affine warping of a the specified quadrangle in the source image to the specified quadrangle in the destination image. The function `nppiWarpAffineQuad` uses the same formulas for pixel mapping as in `nppiWarpAffine` function. The transform coefficients are computed internally. The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

srcQuad Source quadrangle

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

dstQuad Destination quadrangle

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if srcRoi has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if eInterpolation has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.95 `NppStatus nppiWarpAffineQuad_32f_C3R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit float, three channels).

See also:

[nppiWarpAffineQuad_32f_C1R](#)

7.14.2.96 `NppStatus nppiWarpAffineQuad_32f_C4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit float, four channels).

See also:

[nppiWarpAffineQuad_32f_C1R](#)

7.14.2.97 `NppStatus nppiWarpAffineQuad_32f_P3R (const Npp32f * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst[3], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit float, three planes).

See also:

[nppiWarpAffineQuad_32f_C1R](#)

7.14.2.98 `NppStatus nppiWarpAffineQuad_32f_P4R (const Npp32f * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst[4], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit float, four planes).

See also:

[nppiWarpAffineQuad_32f_C1R](#)

7.14.2.99 `NppStatus nppiWarpAffineQuad_32s_AC4R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit signed integer, four channels RGBA).

See also:

[nppiWarpAffineQuad_32s_C1R](#)

7.14.2.100 `NppStatus nppiWarpAffineQuad_32s_C1R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit signed integer, single channel).

This function performs affine warping of a the specified quadrangle in the source image to the specified quadrangle in the destination image. The function `nppiWarpAffineQuad` uses the same formulas for pixel mapping as in `nppiWarpAffine` function. The transform coefficients are computed internally. The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

srcQuad Source quadrangle

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

dstQuad Destination quadrangle

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the `srcRoi` and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if `srcRoi` has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if `eInterpolation` has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment

7.14.2.101 `NppStatus nppiWarpAffineQuad_32s_C3R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit signed integer, three channels).

See also:

[nppiWarpAffineQuad_32s_C1R](#)

7.14.2.102 `NppStatus nppiWarpAffineQuad_32s_C4R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit signed integer, four channels).

See also:

[nppiWarpAffineQuad_32s_C1R](#)

7.14.2.103 `NppStatus nppiWarpAffineQuad_32s_P3R (const Npp32s * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst[3], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit signed integer, three planes).

See also:

[nppiWarpAffineQuad_32s_C1R](#)

7.14.2.104 `NppStatus nppiWarpAffineQuad_32s_P4R (const Npp32s * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst[4], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (32bit signed integer, four planes).

See also:

[nppiWarpAffineQuad_32s_C1R](#)

7.14.2.105 `NppStatus nppiWarpAffineQuad_8u_AC4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (8bit unsigned integer, four channels RGBA).

See also:

[nppiWarpAffineQuad_8u_C1R](#)

7.14.2.106 **NppStatus nppiWarpAffineQuad_8u_C1R** (const Npp8u * *pSrc*, NppiSize *oSrcSize*, int *nSrcStep*, NppiRect *srcRoi*, const double *srcQuad*[4][2], Npp8u * *pDst*, int *nDstStep*, NppiRect *dstRoi*, const double *dstQuad*[4][2], int *eInterpolation*)

Affine transform of an image (8bit unsigned integer, single channel).

This function performs affine warping of a the specified quadrangle in the source image to the specified quadrangle in the destination image. The function `nppiWarpAffineQuad` uses the same formulas for pixel mapping as in `nppiWarpAffine` function. The transform coefficients are computed internally. The transformed part of the source image is resampled using the specified `eInterpolation` method and written to the destination ROI.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but does not perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void*)(pDst + dstRoi.x))` and `(int)((void*)(pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

srcQuad Source quadrangle

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

dstQuad Destination quadrangle

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the `srcRoi` and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if `srcRoi` has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if `eInterpolation` has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI misalignment ignored, internally computed coordinates are used instead

7.14.2.107 `NppStatus nppiWarpAffineQuad_8u_C3R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (8bit unsigned integer, three channels).

See also:

[nppiWarpAffineQuad_8u_C1R](#)

7.14.2.108 `NppStatus nppiWarpAffineQuad_8u_C4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (8bit unsigned integer, four channels).

See also:

[nppiWarpAffineQuad_8u_C1R](#)

7.14.2.109 `NppStatus nppiWarpAffineQuad_8u_P3R (const Npp8u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst[3], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (8bit unsigned integer, three planes).

See also:

[nppiWarpAffineQuad_8u_C1R](#)

7.14.2.110 `NppStatus nppiWarpAffineQuad_8u_P4R (const Npp8u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst[4], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Affine transform of an image (8bit unsigned integer, four planes).

See also:

[nppiWarpAffineQuad_8u_C1R](#)

7.14.2.111 `NppStatus nppiWarpPerspective_16u_AC4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, four channels RGBA).

See also:

[nppiWarpPerspective_16u_C1R](#)

7.14.2.112 **NppStatus nppiWarpPerspective_16u_C1R** (const Npp16u * *pSrc*, NppiSize *oSrcSize*, int *nSrcStep*, NppiRect *srcRoi*, Npp16u * *pDst*, int *nDstStep*, NppiRect *dstRoi*, const double *coeffs*[3][3], int *eInterpolation*)

Perspective transform of an image (16bit unsigned integer, single channel).

This function operates using given transform coefficients that can be obtained by using nppiGetPerspectiveTransform function or set explicitly. The function operates on source and destination regions of interest. The perspective warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$X_{new} = \frac{C_{00} * x + C_{01} * y + C_{02}}{C_{20} * x + C_{21} * y + C_{22}} \quad Y_{new} = \frac{C_{10} * x + C_{11} * y + C_{12}}{C_{20} * x + C_{21} * y + C_{22}}$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions nppiGetPerspectiveQuad and nppiGetPerspectiveBound can help with destination ROI specification.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but does not perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void*)(pDst + dstRoi.x))` and `(int)((void*)(pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc Source-Image Pointer.

oSrcSize Size of source image in pixels

nSrcStep Source-Image Line Step.

srcRoi Source ROI

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

dstRoi Destination ROI

coeffs Perspective transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- `NPP_RECT_ERROR` Indicates an error condition if width or height of the intersection of the *srcRoi* and source image is less than or equal to 1
- `NPP_WRONG_INTERSECTION_ROI_ERROR` Indicates an error condition if *srcRoi* has no intersection with the source image
- `NPP_INTERPOLATION_ERROR` Indicates an error condition if *eInterpolation* has an illegal value
- `NPP_COEFF_ERROR` Indicates an error condition if coefficient values are invalid

- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.113 `NppStatus nppiWarpPerspective_16u_C3R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, three channels).

See also:

[nppiWarpPerspective_16u_C1R](#)

7.14.2.114 `NppStatus nppiWarpPerspective_16u_C4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, four channels).

See also:

[nppiWarpPerspective_16u_C1R](#)

7.14.2.115 `NppStatus nppiWarpPerspective_16u_P3R (const Npp16u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, three planes).

See also:

[nppiWarpPerspective_16u_C1R](#)

7.14.2.116 `NppStatus nppiWarpPerspective_16u_P4R (const Npp16u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, four planes).

See also:

[nppiWarpPerspective_16u_C1R](#)

7.14.2.117 `NppStatus nppiWarpPerspective_32f_AC4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit float, four channels RGBA).

See also:

[nppiWarpPerspective_32f_C1R](#)

7.14.2.118 `NppStatus nppiWarpPerspective_32f_C1R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit float, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetPerspectiveTransform` function or set explicitly. The function operates on source and destination regions of interest. The perspective warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$X_{new} = \frac{C_{00} * x + C_{01} * y + C_{02}}{C_{20} * x + C_{21} * y + C_{22}} \quad Y_{new} = \frac{C_{10} * x + C_{11} * y + C_{12}}{C_{20} * x + C_{21} * y + C_{22}}$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetPerspectiveQuad` and `nppiGetPerspectiveBound` can help with destination ROI specification.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Perspective transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- `NPP_RECT_ERROR` Indicates an error condition if width or height of the intersection of the `srcRoi` and source image is less than or equal to 1
- `NPP_WRONG_INTERSECTION_ROI_ERROR` Indicates an error condition if `srcRoi` has no intersection with the source image
- `NPP_INTERPOLATION_ERROR` Indicates an error condition if `eInterpolation` has an illegal value
- `NPP_COEFF_ERROR` Indicates an error condition if coefficient values are invalid

- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.119 `NppStatus nppiWarpPerspective_32f_C3R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit float, three channels).

See also:

[nppiWarpPerspective_32f_C1R](#)

7.14.2.120 `NppStatus nppiWarpPerspective_32f_C4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit float, four channels).

See also:

[nppiWarpPerspective_32f_C1R](#)

7.14.2.121 `NppStatus nppiWarpPerspective_32f_P3R (const Npp32f * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit float, three planes).

See also:

[nppiWarpPerspective_32f_C1R](#)

7.14.2.122 `NppStatus nppiWarpPerspective_32f_P4R (const Npp32f * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit float, four planes).

See also:

[nppiWarpPerspective_32f_C1R](#)

7.14.2.123 `NppStatus nppiWarpPerspective_32s_AC4R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit signed integer, four channels RGBA).

See also:

[nppiWarpPerspective_32s_C1R](#)

7.14.2.124 `NppStatus nppiWarpPerspective_32s_C1R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit signed integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetPerspectiveTransform` function or set explicitly. The function operates on source and destination regions of interest. The perspective warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$X_{new} = \frac{C_{00} * x + C_{01} * y + C_{02}}{C_{20} * x + C_{21} * y + C_{22}} \quad Y_{new} = \frac{C_{10} * x + C_{11} * y + C_{12}}{C_{20} * x + C_{21} * y + C_{22}}$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetPerspectiveQuad` and `nppiGetPerspectiveBound` can help with destination ROI specification.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Perspective transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- `NPP_RECT_ERROR` Indicates an error condition if width or height of the intersection of the `srcRoi` and source image is less than or equal to 1
- `NPP_WRONG_INTERSECTION_ROI_ERROR` Indicates an error condition if `srcRoi` has no intersection with the source image
- `NPP_INTERPOLATION_ERROR` Indicates an error condition if `eInterpolation` has an illegal value
- `NPP_COEFF_ERROR` Indicates an error condition if coefficient values are invalid

- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.125 `NppStatus nppiWarpPerspective_32s_C3R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit signed integer, three channels).

See also:

[nppiWarpPerspective_32s_C1R](#)

7.14.2.126 `NppStatus nppiWarpPerspective_32s_C4R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit signed integer, four channels).

See also:

[nppiWarpPerspective_32s_C1R](#)

7.14.2.127 `NppStatus nppiWarpPerspective_32s_P3R (const Npp32s * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit signed integer, three planes).

See also:

[nppiWarpPerspective_32s_C1R](#)

7.14.2.128 `NppStatus nppiWarpPerspective_32s_P4R (const Npp32s * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (32bit signed integer, four planes).

See also:

[nppiWarpPerspective_32s_C1R](#)

7.14.2.129 `NppStatus nppiWarpPerspective_8u_AC4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, four channels RGBA).

See also:

[nppiWarpPerspective_8u_C1R](#)

7.14.2.130 `NppStatus nppiWarpPerspective_8u_C1R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetPerspectiveTransform` function or set explicitly. The function operates on source and destination regions of interest. The perspective warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$X_{new} = \frac{C_{00} * x + C_{01} * y + C_{02}}{C_{20} * x + C_{21} * y + C_{22}} \quad Y_{new} = \frac{C_{10} * x + C_{11} * y + C_{12}}{C_{20} * x + C_{21} * y + C_{22}}$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetPerspectiveQuad` and `nppiGetPerspectiveBound` can help with destination ROI specification.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but does not perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void *) (pDst + dstRoi.x))` and `(int)((void *) (pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc Source-Image Pointer.

oSrcSize Size of source image in pixels

nSrcStep Source-Image Line Step.

srcRoi Source ROI

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

dstRoi Destination ROI

coeffs Perspective transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if srcRoi has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if eInterpolation has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.131 `NppStatus nppiWarpPerspective_8u_C3R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, three channels).

See also:

[nppiWarpPerspective_8u_C1R](#)

7.14.2.132 `NppStatus nppiWarpPerspective_8u_C4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, four channels).

See also:

[nppiWarpPerspective_8u_C1R](#)

7.14.2.133 `NppStatus nppiWarpPerspective_8u_P3R (const Npp8u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, three planes).

See also:

[nppiWarpPerspective_8u_C1R](#)

7.14.2.134 `NppStatus nppiWarpPerspective_8u_P4R (const Npp8u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, four planes).

See also:

[nppiWarpPerspective_8u_C1R](#)

7.14.2.135 `NppStatus nppiWarpPerspectiveBack_16u_AC4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (16bit unsigned integer, four channels RGBA).

See also:

[nppiWarpPerspectiveBack_16u_C1R](#)

7.14.2.136 `NppStatus nppiWarpPerspectiveBack_16u_C1R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (16bit unsigned integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetPerspectiveTransform` function or set explicitly. Thus there is no need to invert coefficients in your application before calling `WarpPerspectiveBack`. The function operates on source and destination regions of interest. The perspective warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$\frac{C_{00} * X_{new} + C_{01} * Y_{new} + C_{02}}{C_{20} * X_{new} + C_{21} * Y_{new} + C_{22}} = x \quad \frac{C_{10} * X_{new} + C_{11} * Y_{new} + C_{12}}{C_{20} * X_{new} + C_{21} * Y_{new} + C_{22}} = y$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetPerspectiveQuad` and `nppiGetPerspectiveBound` can help with destination ROI specification.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but does not perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void *) (pDst + dstRoi.x))` and `(int)((void *) (pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc Source-Image Pointer.

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Perspective transform coefficients

eInterpolation Interpolation mode: can be NPPI_INTER_NN, NPPI_INTER_LINEAR or NPPI_INTER_CUBIC

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if srcRoi has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if eInterpolation has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.137 `NppStatus nppiWarpPerspectiveBack_16u_C3R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (16bit unsigned integer, three channels).

See also:

[npippiWarpPerspectiveBack_16u_C1R](#)

7.14.2.138 `NppStatus nppiWarpPerspectiveBack_16u_C4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (16bit unsigned integer, four channels).

See also:

[npippiWarpPerspectiveBack_16u_C1R](#)

7.14.2.139 `NppStatus nppiWarpPerspectiveBack_16u_P3R (const Npp16u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (16bit unsigned integer, three planes).

See also:

[nppiWarpPerspectiveBack_16u_C1R](#)

7.14.2.140 `NppStatus nppiWarpPerspectiveBack_16u_P4R (const Npp16u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp16u * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (16bit unsigned integer, four planes).

See also:

[nppiWarpPerspectiveBack_16u_C1R](#)

7.14.2.141 `NppStatus nppiWarpPerspectiveBack_32f_AC4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (32bit float, four channels RGBA).

See also:

[nppiWarpPerspectiveBack_32f_C1R](#)

7.14.2.142 `NppStatus nppiWarpPerspectiveBack_32f_C1R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (32bit float, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetPerspectiveTransform` function or set explicitly. Thus there is no need to invert coefficients in your application before calling `WarpPerspectiveBack`. The function operates on source and destination regions of interest. The perspective warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$\frac{C_{00} * X_{new} + C_{01} * Y_{new} + C_{02}}{C_{20} * X_{new} + C_{21} * Y_{new} + C_{22}} = x \quad \frac{C_{10} * X_{new} + C_{11} * Y_{new} + C_{12}}{C_{20} * X_{new} + C_{21} * Y_{new} + C_{22}} = y$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetPerspectiveQuad` and `nppiGetPerspectiveBound` can help with destination ROI specification.

Parameters:

pSrc Source-Image Pointer.

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Perspective transform coefficients

eInterpolation Interpolation mode: can be NPPI_INTER_NN, NPPI_INTER_LINEAR or NPPI_INTER_CUBIC

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if srcRoi has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if eInterpolation has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.143 NppStatus npapiWarpPerspectiveBack_32f_C3R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit float, three channels).

See also:

[npapiWarpPerspectiveBack_32f_C1R](#)

7.14.2.144 NppStatus npapiWarpPerspectiveBack_32f_C4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)

Inverse perspective transform of an image (32bit float, four channels).

See also:

[npapiWarpPerspectiveBack_32f_C1R](#)

7.14.2.145 `NppStatus nppiWarpPerspectiveBack_32f_P3R (const Npp32f * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (32bit float, three planes).

See also:

[nppiWarpPerspectiveBack_32f_C1R](#)

7.14.2.146 `NppStatus nppiWarpPerspectiveBack_32f_P4R (const Npp32f * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32f * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (32bit float, four planes).

See also:

[nppiWarpPerspectiveBack_32f_C1R](#)

7.14.2.147 `NppStatus nppiWarpPerspectiveBack_32s_AC4R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (32bit signed integer, four channels RGBA).

See also:

[nppiWarpPerspectiveBack_32s_C1R](#)

7.14.2.148 `NppStatus nppiWarpPerspectiveBack_32s_C1R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (32bit signed integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetPerspectiveTransform` function or set explicitly. Thus there is no need to invert coefficients in your application before calling `WarpPerspectiveBack`. The function operates on source and destination regions of interest. The perspective warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$\frac{C_{00} * X_{new} + C_{01} * Y_{new} + C_{02}}{C_{20} * X_{new} + C_{21} * Y_{new} + C_{22}} = x \quad \frac{C_{10} * X_{new} + C_{11} * Y_{new} + C_{12}}{C_{20} * X_{new} + C_{21} * Y_{new} + C_{22}} = y$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetPerspectiveQuad` and `nppiGetPerspectiveBound` can help with destination ROI specification.

Parameters:

pSrc Source-Image Pointer.

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Perspective transform coefficients

eInterpolation Interpolation mode: can be NPPI_INTER_NN, NPPI_INTER_LINEAR or NPPI_INTER_CUBIC

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if srcRoi has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if eInterpolation has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.149 `NppStatus nppiWarpPerspectiveBack_32s_C3R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (32bit signed integer, three channels).

See also:

[nppiWarpPerspectiveBack_32s_C1R](#)

7.14.2.150 `NppStatus nppiWarpPerspectiveBack_32s_C4R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (32bit signed integer, four channels).

See also:

[nppiWarpPerspectiveBack_32s_C1R](#)

7.14.2.151 `NppStatus nppiWarpPerspectiveBack_32s_P3R (const Npp32s * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (32bit signed integer, three planes).

See also:

[nppiWarpPerspectiveBack_32s_C1R](#)

7.14.2.152 `NppStatus nppiWarpPerspectiveBack_32s_P4R (const Npp32s * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp32s * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (32bit signed integer, four planes).

See also:

[nppiWarpPerspectiveBack_32s_C1R](#)

7.14.2.153 `NppStatus nppiWarpPerspectiveBack_8u_AC4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (8bit unsigned integer, four channels RGBA).

See also:

[nppiWarpPerspectiveBack_8u_C1R](#)

7.14.2.154 `NppStatus nppiWarpPerspectiveBack_8u_C1R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (8bit unsigned integer, single channel).

This function operates using given transform coefficients that can be obtained by using `nppiGetPerspectiveTransform` function or set explicitly. Thus there is no need to invert coefficients in your application before calling `WarpPerspectiveBack`. The function operates on source and destination regions of interest. The perspective warp function transforms the source image pixel coordinates (x, y) according to the following formulas:

$$\frac{C_{00} * X_{new} + C_{01} * Y_{new} + C_{02}}{C_{20} * X_{new} + C_{21} * Y_{new} + C_{22}} = x \quad \frac{C_{10} * X_{new} + C_{11} * Y_{new} + C_{12}}{C_{20} * X_{new} + C_{21} * Y_{new} + C_{22}} = y$$

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI. The functions `nppiGetPerspectiveQuad` and `nppiGetPerspectiveBound` can help with destination ROI specification.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but does not perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical

stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void*)(pDst + dstRoi.x))` and `(int)((void*)(pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

coeffs Perspective transform coefficients

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the *srcRoi* and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if *srcRoi* has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if *eInterpolation* has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.155 `NppStatus nppiWarpPerspectiveBack_8u_C3R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (8bit unsigned integer, three channels).

See also:

[nppiWarpPerspectiveBack_8u_C1R](#)

7.14.2.156 `NppStatus nppiWarpPerspectiveBack_8u_C4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (8bit unsigned integer, four channels).

See also:

[nppiWarpPerspectiveBack_8u_C1R](#)

7.14.2.157 `NppStatus nppiWarpPerspectiveBack_8u_P3R (const Npp8u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst[3], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (8bit unsigned integer, three planes).

See also:

[nppiWarpPerspectiveBack_8u_C1R](#)

7.14.2.158 `NppStatus nppiWarpPerspectiveBack_8u_P4R (const Npp8u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, Npp8u * pDst[4], int nDstStep, NppiRect dstRoi, const double coeffs[3][3], int eInterpolation)`

Inverse perspective transform of an image (8bit unsigned integer, four planes).

See also:

[nppiWarpPerspectiveBack_8u_C1R](#)

7.14.2.159 `NppStatus nppiWarpPerspectiveQuad_16u_AC4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, four channels RGBA).

See also:

[nppiWarpPerspectiveQuad_16u_C1R](#)

7.14.2.160 `NppStatus nppiWarpPerspectiveQuad_16u_C1R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, single channel).

This function performs perspective warping of a the specified quadrangle in the source image to the specified quadrangle in the destination image. The function `nppiWarpPerspectiveQuad` uses the same formulas for pixel mapping as in `nppiWarpPerspective` function. The transform coefficients are computed internally.

The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but does not perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void *) (pDst + dstRoi.x))` and `(int)((void *) (pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

srcQuad Source quadrangle

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

dstQuad Destination quadrangle

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the *srcRoi* and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if *srcRoi* has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if *eInterpolation* has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.161 `NppStatus nppiWarpPerspectiveQuad_16u_C3R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, three channels).

See also:

[nppiWarpPerspectiveQuad_16u_C1R](#)

7.14.2.162 `NppStatus nppiWarpPerspectiveQuad_16u_C4R (const Npp16u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, four channels).

See also:

[nppiWarpPerspectiveQuad_16u_C1R](#)

7.14.2.163 `NppStatus nppiWarpPerspectiveQuad_16u_P3R (const Npp16u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst[3], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, three planes).

See also:

[nppiWarpPerspectiveQuad_16u_C1R](#)

7.14.2.164 `NppStatus nppiWarpPerspectiveQuad_16u_P4R (const Npp16u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp16u * pDst[4], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (16bit unsigned integer, four planes).

See also:

[nppiWarpPerspectiveQuad_16u_C1R](#)

7.14.2.165 `NppStatus nppiWarpPerspectiveQuad_32f_AC4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit float, four channels RGBA).

See also:

[nppiWarpPerspectiveQuad_32f_C1R](#)

7.14.2.166 `NppStatus nppiWarpPerspectiveQuad_32f_C1R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit float, single channel).

This function performs perspective warping of a the specified quadrangle in the source image to the specified quadrangle in the destination image. The function `nppiWarpPerspectiveQuad` uses the same formulas for pixel mapping as in `nppiWarpPerspective` function. The transform coefficients are computed internally. The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI.

Parameters:

pSrc [Source-Image Pointer](#).

oSrcSize Size of source image in pixels

nSrcStep [Source-Image Line Step](#).

srcRoi Source ROI

srcQuad Source quadrangle

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

dstRoi Destination ROI

dstQuad Destination quadrangle

eInterpolation Interpolation mode: can be `NPPI_INTER_NN`, `NPPI_INTER_LINEAR` or `NPPI_INTER_CUBIC`

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the `srcRoi` and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if `srcRoi` has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if `eInterpolation` has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.167 `NppStatus nppiWarpPerspectiveQuad_32f_C3R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit float, three channels).

See also:

[nppiWarpPerspectiveQuad_32f_C1R](#)

7.14.2.168 `NppStatus nppiWarpPerspectiveQuad_32f_C4R (const Npp32f * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit float, four channels).

See also:

[nppiWarpPerspectiveQuad_32f_C1R](#)

7.14.2.169 `NppStatus nppiWarpPerspectiveQuad_32f_P3R (const Npp32f * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst[3], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit float, three planes).

See also:

[nppiWarpPerspectiveQuad_32f_C1R](#)

7.14.2.170 `NppStatus nppiWarpPerspectiveQuad_32f_P4R (const Npp32f * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32f * pDst[4], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit float, four planes).

See also:

[nppiWarpPerspectiveQuad_32f_C1R](#)

7.14.2.171 `NppStatus nppiWarpPerspectiveQuad_32s_AC4R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit signed integer, four channels RGBA).

See also:

[nppiWarpPerspectiveQuad_32s_C1R](#)

7.14.2.172 `NppStatus nppiWarpPerspectiveQuad_32s_C1R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit signed integer, single channel).

This function performs perspective warping of a the specified quadrangle in the source image to the specified quadrangle in the destination image. The function `nppiWarpPerspectiveQuad` uses the same formulas for pixel mapping as in `nppiWarpPerspective` function. The transform coefficients are computed internally. The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI.

Parameters:

pSrc [Source-Image Pointer](#).
oSrcSize Size of source image in pixels
nSrcStep [Source-Image Line Step](#).
srcRoi Source ROI
srcQuad Source quadrangle
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
dstRoi Destination ROI
dstQuad Destination quadrangle
eInterpolation Interpolation mode: can be NPPI_INTER_NN, NPPI_INTER_LINEAR or NPPI_INTER_CUBIC

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if srcRoi has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if eInterpolation has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.173 `NppStatus nppiWarpPerspectiveQuad_32s_C3R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit signed integer, three channels).

See also:

[npippiWarpPerspectiveQuad_32s_C1R](#)

7.14.2.174 `NppStatus nppiWarpPerspectiveQuad_32s_C4R (const Npp32s * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit signed integer, four channels).

See also:

[npippiWarpPerspectiveQuad_32s_C1R](#)

7.14.2.175 `NppStatus nppiWarpPerspectiveQuad_32s_P3R (const Npp32s * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst[3], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit signed integer, three planes).

See also:

[nppiWarpPerspectiveQuad_32s_C1R](#)

7.14.2.176 `NppStatus nppiWarpPerspectiveQuad_32s_P4R (const Npp32s * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp32s * pDst[4], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (32bit signed integer, four planes).

See also:

[nppiWarpPerspectiveQuad_32s_C1R](#)

7.14.2.177 `NppStatus nppiWarpPerspectiveQuad_8u_AC4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, four channels RGBA).

See also:

[nppiWarpPerspectiveQuad_8u_C1R](#)

7.14.2.178 `NppStatus nppiWarpPerspectiveQuad_8u_C1R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, single channel).

This function performs perspective warping of a the specified quadrangle in the source image to the specified quadrangle in the destination image. The function `nppiWarpPerspectiveQuad` uses the same formulas for pixel mapping as in `nppiWarpPerspective` function. The transform coefficients are computed internally. The transformed part of the source image is resampled using the specified interpolation method and written to the destination ROI.

NPPI specific recommendation: The function operates using 2 types of kernels: fast and accurate. The fast method is about 4 times faster than its accurate variant, but does not perform memory access checks and requires the destination ROI to be 64 bytes aligned. Hence any destination ROI is chunked into 3 vertical stripes: the first and the third are processed by accurate kernels and the central one is processed by the fast one. In order to get the maximum available speed of execution, the projection of destination ROI onto image addresses must be 64 bytes aligned. This is always true if the values `(int)((void *) (pDst + dstRoi.x))` and `(int)((void *) (pDst + dstRoi.x + dstRoi.width))` are multiples of 64. Another rule of thumb is to specify destination ROI in such way that left and right sides of the projected image are separated from the ROI by at least 63 bytes from each side. However, this requires the whole ROI to be part of allocated memory. In case when the conditions above are not satisfied, the function may decrease in speed slightly and will return `NPP_MISALIGNED_DST_ROI_WARNING` warning.

Parameters:

pSrc [Source-Image Pointer](#).
oSrcSize Size of source image in pixels
nSrcStep [Source-Image Line Step](#).
srcRoi Source ROI
srcQuad Source quadrangle
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
dstRoi Destination ROI
dstQuad Destination quadrangle
eInterpolation Interpolation mode: can be NPPI_INTER_NN, NPPI_INTER_LINEAR or NPPI_INTER_CUBIC

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_RECT_ERROR](#) Indicates an error condition if width or height of the intersection of the srcRoi and source image is less than or equal to 1
- [NPP_WRONG_INTERSECTION_ROI_ERROR](#) Indicates an error condition if srcRoi has no intersection with the source image
- [NPP_INTERPOLATION_ERROR](#) Indicates an error condition if eInterpolation has an illegal value
- [NPP_COEFF_ERROR](#) Indicates an error condition if coefficient values are invalid
- [NPP_WRONG_INTERSECTION_QUAD_WARNING](#) Indicates a warning that no operation is performed if the transformed source ROI has no intersection with the destination ROI
- [NPP_MISALIGNED_DST_ROI_WARNING](#) Indicates a warning that the speed of primitive execution was reduced due to destination ROI

7.14.2.179 `NppStatus nppiWarpPerspectiveQuad_8u_C3R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, three channels).

See also:

[npippiWarpPerspectiveQuad_8u_C1R](#)

7.14.2.180 `NppStatus nppiWarpPerspectiveQuad_8u_C4R (const Npp8u * pSrc, NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst, int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, four channels).

See also:

[npippiWarpPerspectiveQuad_8u_C1R](#)

7.14.2.181 `NppStatus nppiWarpPerspectiveQuad_8u_P3R (const Npp8u * pSrc[3], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst[3], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, three planes).

See also:

[nppiWarpPerspectiveQuad_8u_C1R](#)

7.14.2.182 `NppStatus nppiWarpPerspectiveQuad_8u_P4R (const Npp8u * pSrc[4], NppiSize oSrcSize, int nSrcStep, NppiRect srcRoi, const double srcQuad[4][2], Npp8u * pDst[4], int nDstStep, NppiRect dstRoi, const double dstQuad[4][2], int eInterpolation)`

Perspective transform of an image (8bit unsigned integer, four planes).

See also:

[nppiWarpPerspectiveQuad_8u_C1R](#)

7.15 Color Conversion

Image color space and sample conversion operations.

RGBToYCbCr

RGB to YCbCr color conversion.

- **NppStatus nppiRGBToYCbCr_8u_C3R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
3 channel 8-bit unsigned packed RGB to packed YCbCr color conversion.
- **NppStatus nppiRGBToYCbCr422_8u_C3C2R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
3 channel 8-bit unsigned RGB to 2 channel chroma packed YCbCr422 color conversion.
- **NppStatus nppiRGBToYCbCr420_8u_C3P3R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** **pDst, int nDstStep[3], **NppiSize** oSizeROI)
3 channel 8-bit unsigned packed RGB to planar YCbCr420 color conversion.
- **NppStatus nppiRGBToYCbCr_8u_P3R** (const **Npp8u** *const *pSrc, int nSrcStep, **Npp8u** **pDst, int nDstStep, **NppiSize** oSizeROI)
3 channel planar 8-bit unsigned RGB to YCbCr color conversion.
- **NppStatus nppiRGBToYCbCr_8u_AC4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 8-bit unsigned RGB to YCbCr color conversion, ignoring Alpha.

YCbCrToRGB

YCbCr to RGB color conversion.

- **NppStatus nppiYCbCrToRGB_8u_C3R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
3 channel 8-bit unsigned packed YCbCr to RGB color conversion.
- **NppStatus nppiYCbCrToRGB_8u_P3R** (const **Npp8u** *const *pSrc, int nSrcStep, **Npp8u** **pDst, int nDstStep, **NppiSize** oSizeROI)
3 channel 8-bit unsigned planar YCbCr to RGB color conversion.
- **NppStatus nppiYCbCrToRGB_8u_AC4R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
4 channel 8-bit unsigned packed YCbCr to RGB color conversion, not affecting Alpha.
- **NppStatus nppiYCbCr422ToRGB_8u_C2C3R** (const **Npp8u** *pSrc, int nSrcStep, **Npp8u** *pDst, int nDstStep, **NppiSize** oSizeROI)
2 channel 8-bit unsigned YCbCr422 to 3 channel packed RGB color conversion.

- `NppStatus nppiYCbCr420ToRGB_8u_P3C3R` (const `Npp8u` *const *pSrc, int nSrcStep[3], `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI)

3 channel 8-bit unsigned planar YCbCr420 to packed RGB color conversion.

Sample Pattern Conversion.

- `NppStatus nppiYCbCr422ToYCbCr420_8u_P3R` (const `Npp8u` *const *pSrc, int nSrcStep[3], `Npp8u` **pDst, int nDstStep[3], `NppiSize` oSizeROI)

3 channel 8-bit unsigned planar YCbCr:422 to YCbCr:420 resampling.

- `NppStatus nppiYCbCr422ToYCbCr411_8u_P3R` (const `Npp8u` *const *pSrc, int nSrcStep[3], `Npp8u` **pDst, int nDstStep[3], `NppiSize` oSizeROI)

3 channel 8-bit unsigned planar YCbCr:422 to YCbCr:411 resampling.

- `NppStatus nppiYCbCr420ToYCbCr422_8u_P3R` (const `Npp8u` *const *pSrc, int nSrcStep[3], `Npp8u` **pDst, int nDstStep[3], `NppiSize` oSizeROI)

3 channel 8-bit unsigned planar YCbCr:420 to YCbCr:422 resampling.

- `NppStatus nppiYCbCr420ToYCbCr411_8u_P3P2R` (const `Npp8u` *const *pSrc, int aSrcStep[3], `Npp8u` *pDstY, int nDstYStep, `Npp8u` *pDstCbCr, int nDstCbCrStep, `NppiSize` oSizeROI)

3 channel 8-bit unsigned planar YCbCr:420 to YCbCr:411 resampling.

Color Processing

Color manipulation functions.

- `NppStatus nppiColorTwist32f_8u_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp32f` twist[3][4])

3 channel 8-bit unsigned color twist.

- `NppStatus nppiColorTwist32f_8u_P3R` (const `Npp8u` *const *pSrc, int nSrcStep, `Npp8u` **pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp32f` twist[3][4])

3 channel planar 8-bit unsigned color twist.

- `NppStatus nppiColorTwist32f_8u_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp32f` twist[3][4])

4 channel 8-bit unsigned color twist, not affecting Alpha.

- `NppStatus nppiLUT_Linear_8u_C1R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp32s` *pValues, const `Npp32s` *pLevels, int nLevels)

8-bit unsigned look-up-table color conversion.

- `NppStatus nppiLUT_Linear_8u_C3R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp32s` *pValues[3], const `Npp32s` *pLevels[3], int nLevels[3])

3 channel 8-bit unsigned look-up-table color conversion.

- `NppStatus nppiLUT_Linear_8u_AC4R` (const `Npp8u` *pSrc, int nSrcStep, `Npp8u` *pDst, int nDstStep, `NppiSize` oSizeROI, const `Npp32s` *pValues[4], const `Npp32s` *pLevels[4], int nLevels[4])

4 channel 8-bit unsigned look-up-table color conversion, not affecting Alpha.

7.15.1 Detailed Description

Image color space and sample conversion operations.

7.15.2 Function Documentation

7.15.2.1 **NppStatus nppiColorTwist32f_8u_AC4R** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp32f *twist*[3][4])

4 channel 8-bit unsigned color twist, not affecting Alpha.

An input color twist matrix with floating-point pixel values is applied with in ROI. Alpha channel is the last channel and is not processed.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

twist The color twist matrix with floating-point pixel values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.2.2 **NppStatus nppiColorTwist32f_8u_C3R** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp32f *twist*[3][4])

3 channel 8-bit unsigned color twist.

An input color twist matrix with floating-point pixel values is applied within ROI.

Parameters:

pSrc [Source-Image Pointer](#).

nSrcStep [Source-Image Line Step](#).

pDst [Destination-Image Pointer](#).

nDstStep [Destination-Image Line Step](#).

oSizeROI [Region-of-Interest \(ROI\)](#).

twist The color twist matrix with floating-point pixel values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.2.3 **NppStatus nppiColorTwist32f_8u_P3R** (const Npp8u *const *pSrc, int nSrcStep, Npp8u **pDst, int nDstStep, NppiSize oSizeROI, const Npp32f twist[3][4])

3 channel planar 8-bit unsigned color twist.

An input color twist matrix with floating-point pixel values is applied within ROI.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

twist The color twist matrix with floating-point pixel values.

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.2.4 **NppStatus nppiLUT_Linear_8u_AC4R** (const Npp8u *pSrc, int nSrcStep, Npp8u *pDst, int nDstStep, NppiSize oSizeROI, const Npp32s *pValues[4], const Npp32s *pLevels[4], int nLevels[4])

4 channel 8-bit unsigned look-up-table color conversion, not affecting Alpha.

The LUT is derived from a set of user defined mapping points through linear interpolation. Alpha channel is the last channel and is not processed.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

pValues Double pointer to an [4] of arrays of user defined OUTPUT values per CHANNEL

pLevels Double pointer to an [4] of arrays of user defined INPUT values per CHANNEL

nLevels A [4] array of user defined input/output mapping points (levels) per CHANNEL

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

- [NPP_LUT_NUMBER_OF_LEVELS_ERROR](#) if the number of levels is less than 2.

7.15.2.5 NppStatus nppiLUT_Linear_8u_C1R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp32s * *pValues*, const Npp32s * *pLevels*, int *nLevels*)

8-bit unsigned look-up-table color conversion.

The LUT is derived from a set of user defined mapping points through linear interpolation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).
- pValues* Pointer to an array of user defined OUTPUT values
- pLevels* Pointer to an array of user defined INPUT values
- nLevels* Number of user defined input/output mapping points (levels)

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- [NPP_LUT_NUMBER_OF_LEVELS_ERROR](#) if the number of levels is less than 2.

7.15.2.6 NppStatus nppiLUT_Linear_8u_C3R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*, const Npp32s * *pValues*[3], const Npp32s * *pLevels*[3], int *nLevels*[3])

3 channel 8-bit unsigned look-up-table color conversion.

The LUT is derived from a set of user defined mapping points through linear interpolation.

Parameters:

- pSrc* Source-Image Pointer.
- nSrcStep* Source-Image Line Step.
- pDst* Destination-Image Pointer.
- nDstStep* Destination-Image Line Step.
- oSizeROI* Region-of-Interest (ROI).
- pValues* Double pointer to an [3] of arrays of user defined OUTPUT values per CHANNEL
- pLevels* Double pointer to an [3] of arrays of user defined INPUT values per CHANNEL
- nLevels* A [3] array of user defined input/output mapping points (levels) per CHANNEL

Returns:

Image Data Related Error Codes, ROI Related Error Codes

- [NPP_LUT_NUMBER_OF_LEVELS_ERROR](#) if the number of levels is less than 2.

7.15.2.7 **NppStatus nppiRGBToYCbCr420_8u_C3P3R** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*[3], NppiSize *oSizeROI*)

3 channel 8-bit unsigned packed RGB to planar YCbCr420 color conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.2.8 **NppStatus nppiRGBToYCbCr422_8u_C3C2R** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel 8-bit unsigned RGB to 2 channel chroma packed YCbCr422 color conversion.
 images.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.2.9 **NppStatus nppiRGBToYCbCr_8u_AC4R** (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

4 channel 8-bit unsigned RGB to YCbCr color conversion, ignoring Alpha.
 Alpha channel is the last channel and is not processed.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.2.10 NppStatus nppiRGBToYCbCr_8u_C3R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel 8-bit unsigned packed RGB to packed YCbCr color conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.2.11 NppStatus nppiRGBToYCbCr_8u_P3R (const Npp8u *const * *pSrc*, int *nSrcStep*, Npp8u ** *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel planar 8-bit unsigned RGB to YCbCr color conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.2.12 NppStatus nppiYCbCr420ToRGB_8u_P3C3R (const Npp8u *const * *pSrc*, int *nSrcStep*[3], Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel 8-bit unsigned planar YCbCr420 to packed RGB color conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.2.13 `NppStatus nppiYCbCr420ToYCbCr411_8u_P3P2R (const Npp8u *const *pSrc, int aSrcStep[3], Npp8u *pDstY, int nDstYStep, Npp8u *pDstCbCr, int nDstCbCrStep, NppiSize oSizeROI)`

3 channel 8-bit unsigned planar YCbCr:420 to YCbCr:411 resampling.

Parameters:

pSrc Array of pointers to the source image planes.
aSrcStep Array with distances in bytes between starts of consecutive lines of the source image planes.
pDstY [Destination-Image Pointer](#). Y-channel.
nDstYStep [Destination-Image Line Step](#). Y-channel.
pDstCbCr [Destination-Image Pointer](#). CbCr image.
nDstCbCrStep [Destination-Image Line Step](#). CbCr image.
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.2.14 `NppStatus nppiYCbCr420ToYCbCr422_8u_P3R (const Npp8u *const *pSrc, int nSrcStep[3], Npp8u **pDst, int nDstStep[3], NppiSize oSizeROI)`

3 channel 8-bit unsigned planar YCbCr:420 to YCbCr:422 resampling.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).
nDstStep [Destination-Image Line Step](#).
oSizeROI [Region-of-Interest \(ROI\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.15.2.15 `NppStatus nppiYCbCr422ToRGB_8u_C2C3R (const Npp8u *pSrc, int nSrcStep, Npp8u *pDst, int nDstStep, NppiSize oSizeROI)`

2 channel 8-bit unsigned YCbCr422 to 3 channel packed RGB color conversion.
 images.

Parameters:

pSrc [Source-Image Pointer](#).
nSrcStep [Source-Image Line Step](#).
pDst [Destination-Image Pointer](#).

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.2.16 `NppStatus nppiYCbCr422ToYCbCr411_8u_P3R (const Npp8u *const *pSrc, int nSrcStep[3], Npp8u **pDst, int nDstStep[3], NppiSize oSizeROI)`

3 channel 8-bit unsigned planar YCbCr:422 to YCbCr:411 resampling.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.2.17 `NppStatus nppiYCbCr422ToYCbCr420_8u_P3R (const Npp8u *const *pSrc, int nSrcStep[3], Npp8u **pDst, int nDstStep[3], NppiSize oSizeROI)`

3 channel 8-bit unsigned planar YCbCr:422 to YCbCr:420 resampling.

Parameters:

pSrc Source-Image Pointer.

nSrcStep Source-Image Line Step.

pDst Destination-Image Pointer.

nDstStep Destination-Image Line Step.

oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.2.18 `NppStatus nppiYCbCrToRGB_8u_AC4R (const Npp8u *pSrc, int nSrcStep, Npp8u *pDst, int nDstStep, NppiSize oSizeROI)`

4 channel 8-bit unsigned packed YCbCr to RGB color conversion, not affecting Alpha.

Alpha channel is the last channel and is not processed.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.2.19 NppStatus nppiYCbCrToRGB_8u_C3R (const Npp8u * *pSrc*, int *nSrcStep*, Npp8u * *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel 8-bit unsigned packed YCbCr to RGB color conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.15.2.20 NppStatus nppiYCbCrToRGB_8u_P3R (const Npp8u *const * *pSrc*, int *nSrcStep*, Npp8u ** *pDst*, int *nDstStep*, NppiSize *oSizeROI*)

3 channel 8-bit unsigned planar YCbCr to RGB color conversion.

Parameters:

pSrc Source-Image Pointer.
nSrcStep Source-Image Line Step.
pDst Destination-Image Pointer.
nDstStep Destination-Image Line Step.
oSizeROI Region-of-Interest (ROI).

Returns:

Image Data Related Error Codes, ROI Related Error Codes

7.16 Labeling and Segmentation

Pixel labeling and image segmentation operations.

Typedefs

- typedef struct [NppiGraphcutState](#) [NppiGraphcutState](#)

Functions

- [NppStatus](#) [nppiGraphcutGetSize](#) ([NppiSize](#) oSize, int *pBufSize)
Calculates the size of the temporary buffer for graph-cut with 4 neighborhood labeling.
- [NppStatus](#) [nppiGraphcut8GetSize](#) ([NppiSize](#) oSize, int *pBufSize)
Calculates the size of the temporary buffer for graph-cut with 8 neighborhood labeling.
- [NppStatus](#) [nppiGraphcutInitAlloc](#) ([NppiSize](#) oSize, [NppiGraphcutState](#) **ppState, [Npp8u](#) *pDeviceMem)
Initializes graph-cut state structure and allocates additional resources for graph-cut with 8 neighborhood labeling.
- [NppStatus](#) [nppiGraphcut8InitAlloc](#) ([NppiSize](#) oSize, [NppiGraphcutState](#) **ppState, [Npp8u](#) *pDeviceMem)
Allocates and initializes the graph-cut state structure and additional resources for graph-cut with 8 neighborhood labeling.
- [NppStatus](#) [nppiGraphcutFree](#) ([NppiGraphcutState](#) *pState)
Frees the additional resources of the graph-cut state structure.
- [NppStatus](#) [nppiGraphcut_32s8u](#) ([Npp32s](#) *pTerminals, [Npp32s](#) *pLeftTransposed, [Npp32s](#) *pRightTransposed, [Npp32s](#) *pTop, [Npp32s](#) *pBottom, int nStep, int nTransposedStep, [NppiSize](#) size, [Npp8u](#) *pLabel, int nLabelStep, [NppiGraphcutState](#) *pState)
Graphcut of a flow network (32bit signed integer edge capacities).
- [NppStatus](#) [nppiGraphcut8_32s8u](#) ([Npp32s](#) *pTerminals, [Npp32s](#) *pLeftTransposed, [Npp32s](#) *pRightTransposed, [Npp32s](#) *pTop, [Npp32s](#) *pTopLeft, [Npp32s](#) *pTopRight, [Npp32s](#) *pBottom, [Npp32s](#) *pBottomLeft, [Npp32s](#) *pBottomRight, int nStep, int nTransposedStep, [NppiSize](#) size, [Npp8u](#) *pLabel, int nLabelStep, [NppiGraphcutState](#) *pState)
Graphcut of a flow network (32bit signed integer edge capacities).

7.16.1 Detailed Description

Pixel labeling and image segmentation operations.

7.16.2 Typedef Documentation

7.16.2.1 typedef struct NppiGraphcutState NppiGraphcutState

7.16.3 Function Documentation

7.16.3.1 NppStatus nppiGraphcut8_32s8u (Npp32s * *pTerminals*, Npp32s * *pLeftTransposed*, Npp32s * *pRightTransposed*, Npp32s * *pTop*, Npp32s * *pTopLeft*, Npp32s * *pTopRight*, Npp32s * *pBottom*, Npp32s * *pBottomLeft*, Npp32s * *pBottomRight*, int *nStep*, int *nTransposedStep*, NppiSize *size*, Npp8u * *pLabel*, int *nLabelStep*, NppiGraphcutState * *pState*)

Graphcut of a flow network (32bit signed integer edge capacities).

The function computes the minimal cut (graphcut) of a 2D regular 8-connected graph. The inputs are the capacities of the horizontal (in transposed form), vertical and terminal (source and sink) edges. The capacities to source and sink are stored as capacity differences in the terminals array (*terminals*(*x*) = *source*(*x*) - *sink*(*x*)). The implementation assumes that the edge capacities for boundary edges that would connect to nodes outside the specified domain are set to 0 (for example *left*(0,*) == 0). If this is not fulfilled the computed labeling may be wrong! The computed binary labeling is encoded as unsigned 8bit values (0 / 255).

See also:

[nppiGraphcut8InitAlloc\(\)](#), [nppiGraphcutFree\(\)](#), [nppiGraphcut8GetSize\(\)](#).

Parameters:

- pTerminals* Pointer to differences of terminal edge capacities (*terminal*(*x*) = *source*(*x*) - *sink*(*x*))
- pLeftTransposed* Pointer to transposed left edge capacities (*left*(0,*) must be 0)
- pRightTransposed* Pointer to transposed right edge capacities (*right*(*width*-1,*) must be 0)
- pTop* Pointer to top edge capacities (*top*(*,0) must be 0)
- pTopLeft* Pointer to top left edge capacities (*topleft*(*,0) & *topleft*(0,*) must be 0)
- pTopRight* Pointer to top right edge capacities (*topright*(*,0) & *topright*(*width*-1,*) must be 0)
- pBottom* Pointer to bottom edge capacities (*bottom*(*,*height*-1) must be 0)
- pBottomLeft* Pointer to bottom left edge capacities (*bottomleft*(*,*height*-1) && *bottomleft*(0,*) must be 0)
- pBottomRight* Pointer to bottom right edge capacities (*bottomright*(*,*height*-1) && *bottomright*(*width*-1,*) must be 0)
- nStep* Step in bytes between any pair of sequential rows of edge capacities
- nTransposedStep* Step in bytes between any pair of sequential rows of transposed edge capacities
- size* Graph size
- pLabel* Pointer to destination label image
- nLabelStep* Step in bytes between any pair of sequential rows of label image
- pState* Pointer to graph-cut state structure. This structure must be initialized allocated and initialized using [nppiGraphcut8InitAlloc\(\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.3.2 NppStatus nppiGraphcut8GetSize (NppiSize oSize, int * pBufSize)

Calculates the size of the temporary buffer for graph-cut with 8 neighborhood labeling.

See also:

[nppiGraphcut8InitAlloc\(\)](#), [nppiGraphcut8_32s8u\(\)](#).

Parameters:

oSize Graph size.

pBufSize Pointer to variable that returns the size of the temporary buffer.

Returns:

NPP_SUCCESS Indicates no error. Any other value indicates an error or a warning

NPP_SIZE_ERROR Indicates an error condition if any image dimension has zero or negative value

NPP_NULL_POINTER_ERROR Indicates an error condition if pBufSize pointer is NULL

7.16.3.3 NppStatus nppiGraphcut8InitAlloc (NppiSize oSize, NppiGraphcutState ** ppState, Npp8u * pDeviceMem)

Allocates and initializes the graph-cut state structure and additional resources for graph-cut with 8 neighborhood labeling.

See also:

[nppiGraphcut8_32s8u\(\)](#), [nppiGraphcut8GetSize\(\)](#).

Parameters:

oSize Graph size

ppState Pointer to pointer to graph-cut state structure.

pDeviceMem to the sufficient amount of device memory. The CUDA runtime or NPP memory allocators must be used to allocate this memory. The minimum amount of device memory required to run graph-cut on a for a specific image size is computed by [nppiGraphcut8GetSize\(\)](#).

Returns:

NPP_SUCCESS Indicates no error. Any other value indicates an error or a warning

NPP_SIZE_ERROR Indicates an error condition if any image dimension has zero or negative value

NPP_NULL_POINTER_ERROR Indicates an error condition if pBufSize pointer is NULL

7.16.3.4 NppStatus nppiGraphcut_32s8u (Npp32s * pTerminals, Npp32s * pLeftTransposed, Npp32s * pRightTransposed, Npp32s * pTop, Npp32s * pBottom, int nStep, int nTransposedStep, NppiSize size, Npp8u * pLabel, int nLabelStep, NppiGraphcutState * pState)

Graphcut of a flow network (32bit signed integer edge capacities).

The function computes the minimal cut (graphcut) of a 2D regular 4-connected graph. The inputs are the capacities of the horizontal (in transposed form), vertical and terminal (source and sink) edges. The capacities to source and sink are stored as capacity differences in the terminals array (terminals(x) =

source(x) - sink(x)). The implementation assumes that the edge capacities for boundary edges that would connect to nodes outside the specified domain are set to 0 (for example left(0,*) == 0). If this is not fulfilled the computed labeling may be wrong! The computed binary labeling is encoded as unsigned 8bit values (0 / 255).

See also:

[nppiGraphcutInitAlloc\(\)](#), [nppiGraphcutFree\(\)](#), [nppiGraphcutGetSize\(\)](#).

Parameters:

pTerminals Pointer to differences of terminal edge capacities (terminal(x) = source(x) - sink(x))
pLeftTransposed Pointer to transposed left edge capacities (left(0,*) must be 0)
pRightTransposed Pointer to transposed right edge capacities (right(width-1,*) must be 0)
pTop Pointer to top edge capacities (top(*,0) must be 0)
pBottom Pointer to bottom edge capacities (bottom(*,height-1) must be 0)
nStep Step in bytes between any pair of sequential rows of edge capacities
nTransposedStep Step in bytes between any pair of sequential rows of transposed edge capacities
size Graph size
pLabel Pointer to destination label image
nLabelStep Step in bytes between any pair of sequential rows of label image
pState Pointer to graph-cut state structure. This structure must be initialized allocated and initialized using [nppiGraphcutInitAlloc\(\)](#).

Returns:

[Image Data Related Error Codes](#), [ROI Related Error Codes](#)

7.16.3.5 NppStatus nppiGraphcutFree (NppiGraphcutState * pState)

Frees the additional resources of the graph-cut state structure.

See also:

[nppiGraphcutInitAlloc](#)
[nppiGraphcut8InitAlloc](#)

Parameters:

pState Pointer to graph-cut state structure.

Returns:

NPP_SUCCESS Indicates no error. Any other value indicates an error or a warning
 NPP_SIZE_ERROR Indicates an error condition if any image dimension has zero or negative value
 NPP_NULL_POINTER_ERROR Indicates an error condition if pState pointer is NULL

7.16.3.6 NppStatus nppiGraphcutGetSize (NppiSize *oSize*, int * *pBufSize*)

Calculates the size of the temporary buffer for graph-cut with 4 neighborhood labeling.

See also:

[nppiGraphcutInitAlloc\(\)](#), [nppiGraphcut_32s8u\(\)](#).

Parameters:

oSize Graph size.

pBufSize Pointer to variable that returns the size of the temporary buffer.

Returns:

NPP_SUCCESS Indicates no error. Any other value indicates an error or a warning

NPP_SIZE_ERROR Indicates an error condition if any image dimension has zero or negative value

NPP_NULL_POINTER_ERROR Indicates an error condition if pBufSize pointer is NULL

7.16.3.7 NppStatus nppiGraphcutInitAlloc (NppiSize *oSize*, NppiGraphcutState ** *ppState*, Npp8u * *pDeviceMem*)

Initializes graph-cut state structure and allocates additional resources for graph-cut with 8 neighborhood labeling.

See also:

[nppiGraphcut_32s8u\(\)](#), [nppiGraphcutGetSize\(\)](#).

Parameters:

oSize Graph size

ppState Pointer to pointer to graph-cut state structure.

pDeviceMem pDeviceMem to the sufficient amount of device memory. The CUDA runtime or NPP memory allocators must be used to allocate this memory. The minimum amount of device memory required to run graph-cut on a for a specific image size is computed by [nppiGraphcutGetSize\(\)](#).

Returns:

NPP_SUCCESS Indicates no error. Any other value indicates an error or a warning

NPP_SIZE_ERROR Indicates an error condition if any image dimension has zero or negative value

NPP_NULL_POINTER_ERROR Indicates an error condition if pBufSize pointer is NULL

7.17 NPP Signal Processing

Memory Allocation

Signal-allocator methods for allocating 1D arrays of data in device memory.

All allocators have size parameters to specify the size of the signal (1D array) being allocated.

The allocator methods return a pointer to the newly allocated memory of appropriate type. If device-memory allocation is not possible due to resource constraints the allocators return 0 (i.e. NULL pointer).

All signal allocators allocate memory aligned such that it is beneficial to the performance of the majority of the signal-processing primitives. It is no mandatory however to use these allocators. Any valid CUDA device-memory pointers can be passed to NPP primitives.

- `Npp8u * nppsMalloc_8u` (int nSize)
8-bit unsigned signal allocator.
- `Npp16u * nppsMalloc_16u` (int nSize)
16-bit unsigned signal allocator.
- `Npp16s * nppsMalloc_16s` (int nSize)
16-bit signal allocator.
- `Npp16sc * nppsMalloc_16sc` (int nSize)
16-bit complex-value signal allocator.
- `Npp32u * nppsMalloc_32u` (int nSize)
32-bit unsigned signal allocator.
- `Npp32s * nppsMalloc_32s` (int nSize)
32-bit integer signal allocator.
- `Npp32sc * nppsMalloc_32sc` (int nSize)
32-bit complex integer signal allocator.
- `Npp32f * nppsMalloc_32f` (int nSize)
32-bit float signal allocator.
- `Npp32fc * nppsMalloc_32fc` (int nSize)
32-bit complex float signal allocator.
- `Npp64s * nppsMalloc_64s` (int nSize)
64-bit long integer signal allocator.
- `Npp64sc * nppsMalloc_64sc` (int nSize)
64-bit complex long integer signal allocator.
- `Npp64f * nppsMalloc_64f` (int nSize)
64-bit float (double) signal allocator.
- `Npp64fc * nppsMalloc_64fc` (int nSize)

64-bit complex complex signal allocator.

- void `nppsFree` (void *pValues)
Free method for any 2D allocated memory.

Set

Set methods for 1D vectors of various types.

The copy methods operate on vector data given as a pointer to the underlying data-type (e.g. 8-bit vectors would be passed as pointers to `Npp8u` type) and length of the vectors, i.e. the number of items.

- `NppStatus nppsSet_8u` (`Npp8u` nValue, `Npp8u` *pDst, int nLength)
8-bit unsigned char, vector set method.
- `NppStatus nppsSet_16s` (`Npp16s` nValue, `Npp16s` *pDst, int nLength)
16-bit integer, vector set method.
- `NppStatus nppsSet_16sc` (`Npp16sc` nValue, `Npp16sc` *pDst, int nLength)
16-bit integer complex, vector set method.
- `NppStatus nppsSet_32s` (`Npp32s` nValue, `Npp32s` *pDst, int nLength)
32-bit integer, vector set method.
- `NppStatus nppsSet_32sc` (`Npp32sc` nValue, `Npp32sc` *pDst, int nLength)
32-bit integer complex, vector set method.
- `NppStatus nppsSet_32f` (`Npp32f` nValue, `Npp32f` *pDst, int nLength)
32-bit float, vector set method.
- `NppStatus nppsSet_32fc` (`Npp32fc` nValue, `Npp32fc` *pDst, int nLength)
32-bit float complex, vector set method.
- `NppStatus nppsSet_64s` (`Npp64s` nValue, `Npp64s` *pDst, int nLength)
64-bit long long integer, vector set method.
- `NppStatus nppsSet_64sc` (`Npp64sc` nValue, `Npp64sc` *pDst, int nLength)
64-bit long long integer complex, vector set method.
- `NppStatus nppsSet_64f` (`Npp64f` nValue, `Npp64f` *pDst, int nLength)
64-bit double, vector set method.
- `NppStatus nppsSet_64fc` (`Npp64fc` nValue, `Npp64fc` *pDst, int nLength)
64-bit double complex, vector set method.

Zero

Set signals to zero.

- `NppStatus nppsZero_8u (Npp8u *pDst, int nLength)`
8-bit unsigned char, vector zero method.
- `NppStatus nppsZero_16s (Npp16s *pDst, int nLength)`
16-bit integer, vector zero method.
- `NppStatus nppsZero_16sc (Npp16sc *pDst, int nLength)`
16-bit integer complex, vector zero method.
- `NppStatus nppsZero_32s (Npp32s *pDst, int nLength)`
32-bit integer, vector zero method.
- `NppStatus nppsZero_32sc (Npp32sc *pDst, int nLength)`
32-bit integer complex, vector zero method.
- `NppStatus nppsZero_32f (Npp32f *pDst, int nLength)`
32-bit float, vector zero method.
- `NppStatus nppsZero_32fc (Npp32fc *pDst, int nLength)`
32-bit float complex, vector zero method.
- `NppStatus nppsZero_64s (Npp64s *pDst, int nLength)`
64-bit long long integer, vector zero method.
- `NppStatus nppsZero_64sc (Npp64sc *pDst, int nLength)`
64-bit long long integer complex, vector zero method.
- `NppStatus nppsZero_64f (Npp64f *pDst, int nLength)`
64-bit double, vector zero method.
- `NppStatus nppsZero_64fc (Npp64fc *pDst, int nLength)`
64-bit double complex, vector zero method.

Copy

Copy methods for various type signals.

Copy methods operate on signal data given as a pointer to the underlying data-type (e.g. 8-bit vectors would be passed as pointers to `Npp8u` type) and length of the vectors, i.e. the number of items.

- `NppStatus nppsCopy_8u (const Npp8u *pSrc, Npp8u *pDst, int len)`
8-bit unsigned char, vector copy method
- `NppStatus nppsCopy_16s (const Npp16s *pSrc, Npp16s *pDst, int len)`
16-bit signed short, vector copy method.

- **NppStatus nppsCopy_32s** (const **Npp32s** *pSrc, **Npp32s** *pDst, int nLength)
32-bit signed integer, vector copy method.
- **NppStatus nppsCopy_32f** (const **Npp32f** *pSrc, **Npp32f** *pDst, int len)
32-bit float, vector copy method.
- **NppStatus nppsCopy_64s** (const **Npp64s** *pSrc, **Npp64s** *pDst, int len)
64-bit signed integer, vector copy method.
- **NppStatus nppsCopy_16sc** (const **Npp16sc** *pSrc, **Npp16sc** *pDst, int len)
16-bit complex short, vector copy method.
- **NppStatus nppsCopy_32sc** (const **Npp32sc** *pSrc, **Npp32sc** *pDst, int len)
32-bit complex signed integer, vector copy method.
- **NppStatus nppsCopy_32fc** (const **Npp32fc** *pSrc, **Npp32fc** *pDst, int len)
32-bit complex float, vector copy method.
- **NppStatus nppsCopy_64sc** (const **Npp64sc** *pSrc, **Npp64sc** *pDst, int len)
64-bit complex signed integer, vector copy method.
- **NppStatus nppsCopy_64fc** (const **Npp64fc** *pSrc, **Npp64fc** *pDst, int len)
64-bit complex double, vector copy method.

AddC

Adds a constant value to each sample of a signal.

- **NppStatus nppsAddC_8u_ISfs** (**Npp8u** nValue, **Npp8u** *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char in place signal add constant, scale, then clamp to saturated value
- **NppStatus nppsAddC_8u_Sfs** (const **Npp8u** *pSrc, **Npp8u** nValue, **Npp8u** *pDst, int nLength, int nScaleFactor)
8-bit unsigned charvector add constant, scale, then clamp to saturated value.
- **NppStatus nppsAddC_16u_ISfs** (**Npp16u** nValue, **Npp16u** *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short in place signal add constant, scale, then clamp to saturated value.
- **NppStatus nppsAddC_16u_Sfs** (const **Npp16u** *pSrc, **Npp16u** nValue, **Npp16u** *pDst, int nLength, int nScaleFactor)
16-bit unsigned short vector add constant, scale, then clamp to saturated value.
- **NppStatus nppsAddC_16s_ISfs** (**Npp16s** nValue, **Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short in place signal add constant, scale, then clamp to saturated value.
- **NppStatus nppsAddC_16s_Sfs** (const **Npp16s** *pSrc, **Npp16s** nValue, **Npp16s** *pDst, int nLength, int nScaleFactor)

16-bit signed short signal add constant, scale, then clamp to saturated value.

- **NppStatus nppsAddC_16sc_ISfs** (**Npp16sc** nValue, **Npp16sc** *pSrcDst, int nLength, int nScaleFactor)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal add constant, scale, then clamp to saturated value.

- **NppStatus nppsAddC_16sc_Sfs** (const **Npp16sc** *pSrc, **Npp16sc** nValue, **Npp16sc** *pDst, int nLength, int nScaleFactor)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal add constant, scale, then clamp to saturated value.

- **NppStatus nppsAddC_32s_ISfs** (**Npp32s** nValue, **Npp32s** *pSrcDst, int nLength, int nScaleFactor)

32-bit signed integer in place signal add constant and scale.

- **NppStatus nppsAddC_32s_Sfs** (const **Npp32s** *pSrc, **Npp32s** nValue, **Npp32s** *pDst, int nLength, int nScaleFactor)

32-bit signed integers signal add constant and scale.

- **NppStatus nppsAddC_32sc_ISfs** (**Npp32sc** nValue, **Npp32sc** *pSrcDst, int nLength, int nScaleFactor)

32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal add constant and scale.

- **NppStatus nppsAddC_32sc_Sfs** (const **Npp32sc** *pSrc, **Npp32sc** nValue, **Npp32sc** *pDst, int nLength, int nScaleFactor)

32-bit integer complex number (32 bit real, 32 bit imaginary) signal add constant and scale.

- **NppStatus nppsAddC_32f_I** (**Npp32f** nValue, **Npp32f** *pSrcDst, int nLength)

32-bit floating point in place signal add constant.

- **NppStatus nppsAddC_32f** (const **Npp32f** *pSrc, **Npp32f** nValue, **Npp32f** *pDst, int nLength)

32-bit floating point signal add constant.

- **NppStatus nppsAddC_32fc_I** (**Npp32fc** nValue, **Npp32fc** *pSrcDst, int nLength)

32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal add constant.

- **NppStatus nppsAddC_32fc** (const **Npp32fc** *pSrc, **Npp32fc** nValue, **Npp32fc** *pDst, int nLength)

32-bit floating point complex number (32 bit real, 32 bit imaginary) signal add constant.

- **NppStatus nppsAddC_64f_I** (**Npp64f** nValue, **Npp64f** *pSrcDst, int nLength)

64-bit floating point, in place signal add constant.

- **NppStatus nppsAddC_64f** (const **Npp64f** *pSrc, **Npp64f** nValue, **Npp64f** *pDst, int nLength)

64-bit floating points signal add constant.

- **NppStatus nppsAddC_64fc_I** (**Npp64fc** nValue, **Npp64fc** *pSrcDst, int nLength)

64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal add constant.

- **NppStatus nppsAddC_64fc** (const **Npp64fc** *pSrc, **Npp64fc** nValue, **Npp64fc** *pDst, int nLength)

64-bit floating point complex number (64 bit real, 64 bit imaginary) signal add constant.

AddProductC

Adds product of a constant and each sample of a source signal to the each sample of destination signal.

- **NppStatus** **nppsAddProductC_8u_ISfs** (**Npp8u** nValue, **Npp8u** *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char in place signal add product of signal times constant to destination signal, scale, then clamp to saturated value
- **NppStatus** **nppsAddProductC_8u_Sfs** (const **Npp8u** *pSrc, **Npp8u** nValue, **Npp8u** *pDst, int nLength, int nScaleFactor)
8-bit unsigned char add product of signal times constant to destination signal, scale, then clamp to saturated value.
- **NppStatus** **nppsAddProductC_16u_ISfs** (**Npp16u** nValue, **Npp16u** *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short in place signal add product of signal times constant to destination signal, scale, then clamp to saturated value.
- **NppStatus** **nppsAddProductC_16u_Sfs** (const **Npp16u** *pSrc, **Npp16u** nValue, **Npp16u** *pDst, int nLength, int nScaleFactor)
16-bit unsigned short add product of signal times constant to destination signal, scale, then clamp to saturated value.
- **NppStatus** **nppsAddProductC_16s_ISfs** (**Npp16s** nValue, **Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short in place signal add product of signal times constant to destination signal, scale, then clamp to saturated value.
- **NppStatus** **nppsAddProductC_16s_Sfs** (const **Npp16s** *pSrc, **Npp16s** nValue, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal add product of signal times constant to destination signal, scale, then clamp to saturated value.
- **NppStatus** **nppsAddProductC_16sc_ISfs** (**Npp16sc** nValue, **Npp16sc** *pSrcDst, int nLength, int nScaleFactor)
16-bit integer complex number (16 bit real, 16 bit imaginary)signal add product of signal times constant to destination signal, scale, then clamp to saturated value.
- **NppStatus** **nppsAddProductC_16sc_Sfs** (const **Npp16sc** *pSrc, **Npp16sc** nValue, **Npp16sc** *pDst, int nLength, int nScaleFactor)
16-bit integer complex number (16 bit real, 16 bit imaginary)signal add product of signal times constant to destination signal, scale, then clamp to saturated value.
- **NppStatus** **nppsAddProductC_32s_ISfs** (**Npp32s** nValue, **Npp32s** *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer in place signal add product of signal times constant to destination signal and scale.
- **NppStatus** **nppsAddProductC_32s_Sfs** (const **Npp32s** *pSrc, **Npp32s** nValue, **Npp32s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal add product of signal times constant to destination signal and scale.

- **NppStatus nppsAddProductC_32sc_ISfs** (**Npp32sc** nValue, **Npp32sc** *pSrcDst, int nLength, int nScaleFactor)
32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal add product of signal times constant to destination signal and scale.
- **NppStatus nppsAddProductC_32sc_Sfs** (const **Npp32sc** *pSrc, **Npp32sc** nValue, **Npp32sc** *pDst, int nLength, int nScaleFactor)
32-bit integer complex number (32 bit real, 32 bit imaginary) signal add product of signal times constant to destination signal and scale.
- **NppStatus nppsAddProductC_32f_I** (**Npp32f** nValue, **Npp32f** *pSrcDst, int nLength)
32-bit floating point in place signal add product of signal times constant to destination signal.
- **NppStatus nppsAddProductC_32f** (const **Npp32f** *pSrc, **Npp32f** nValue, **Npp32f** *pDst, int nLength)
32-bit floating point signal add product of signal times constant to destination signal.
- **NppStatus nppsAddProductC_32fc_I** (**Npp32fc** nValue, **Npp32fc** *pSrcDst, int nLength)
32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal add product of signal times constant to destination signal.
- **NppStatus nppsAddProductC_32fc** (const **Npp32fc** *pSrc, **Npp32fc** nValue, **Npp32fc** *pDst, int nLength)
32-bit floating point complex number (32 bit real, 32 bit imaginary) signal add product of signal times constant to destination signal.
- **NppStatus nppsAddProductC_64f_I** (**Npp64f** nValue, **Npp64f** *pSrcDst, int nLength)
64-bit floating point, in place signal add product of signal times constant to destination signal.
- **NppStatus nppsAddProductC_64f** (const **Npp64f** *pSrc, **Npp64f** nValue, **Npp64f** *pDst, int nLength)
64-bit floating point signal add product of signal times constant to destination signal.
- **NppStatus nppsAddProductC_64fc_I** (**Npp64fc** nValue, **Npp64fc** *pSrcDst, int nLength)
64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal add product of signal times constant to destination signal.
- **NppStatus nppsAddProductC_64fc** (const **Npp64fc** *pSrc, **Npp64fc** nValue, **Npp64fc** *pDst, int nLength)
64-bit floating point complex number (64 bit real, 64 bit imaginary) signal add product of signal times constant to destination signal.

MulC

Multiplies each sample of a signal by a constant value.

- **NppStatus nppsMulC_8u_ISfs** (**Npp8u** nValue, **Npp8u** *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char in place signal times constant, scale, then clamp to saturated value

- **NppStatus nppsMulC_8u_Sfs** (const **Npp8u** *pSrc, **Npp8u** nValue, **Npp8u** *pDst, int nLength, int nScaleFactor)
8-bit unsigned char signal times constant, scale, then clamp to saturated value.
- **NppStatus nppsMulC_16u_ISfs** (**Npp16u** nValue, **Npp16u** *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short in place signal times constant, scale, then clamp to saturated value.
- **NppStatus nppsMulC_16u_Sfs** (const **Npp16u** *pSrc, **Npp16u** nValue, **Npp16u** *pDst, int nLength, int nScaleFactor)
16-bit unsigned short signal times constant, scale, then clamp to saturated value.
- **NppStatus nppsMulC_16s_ISfs** (**Npp16s** nValue, **Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short in place signal times constant, scale, then clamp to saturated value.
- **NppStatus nppsMulC_16s_Sfs** (const **Npp16s** *pSrc, **Npp16s** nValue, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal times constant, scale, then clamp to saturated value.
- **NppStatus nppsMulC_16sc_ISfs** (**Npp16sc** nValue, **Npp16sc** *pSrcDst, int nLength, int nScaleFactor)
16-bit integer complex number (16 bit real, 16 bit imaginary) signal times constant, scale, then clamp to saturated value.
- **NppStatus nppsMulC_16sc_Sfs** (const **Npp16sc** *pSrc, **Npp16sc** nValue, **Npp16sc** *pDst, int nLength, int nScaleFactor)
16-bit integer complex number (16 bit real, 16 bit imaginary) signal times constant, scale, then clamp to saturated value.
- **NppStatus nppsMulC_32s_ISfs** (**Npp32s** nValue, **Npp32s** *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer in place signal times constant and scale.
- **NppStatus nppsMulC_32s_Sfs** (const **Npp32s** *pSrc, **Npp32s** nValue, **Npp32s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal times constant and scale.
- **NppStatus nppsMulC_32sc_ISfs** (**Npp32sc** nValue, **Npp32sc** *pSrcDst, int nLength, int nScaleFactor)
32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal times constant and scale.
- **NppStatus nppsMulC_32sc_Sfs** (const **Npp32sc** *pSrc, **Npp32sc** nValue, **Npp32sc** *pDst, int nLength, int nScaleFactor)
32-bit integer complex number (32 bit real, 32 bit imaginary) signal times constant and scale.
- **NppStatus nppsMulC_32f_I** (**Npp32f** nValue, **Npp32f** *pSrcDst, int nLength)
32-bit floating point in place signal times constant.
- **NppStatus nppsMulC_32f** (const **Npp32f** *pSrc, **Npp32f** nValue, **Npp32f** *pDst, int nLength)
32-bit floating point signal times constant.

- `NppStatus nppsMulC_Low_32f16s` (const `Npp32f` *pSrc, `Npp32f` nValue, `Npp16s` *pDst, int nLength)
32-bit floating point signal times constant with output converted to 16-bit signed integer.
- `NppStatus nppsMulC_32f16s_Sfs` (const `Npp32f` *pSrc, `Npp32f` nValue, `Npp16s` *pDst, int nLength, int nScaleFactor)
32-bit floating point signal times constant with output converted to 16-bit signed integer with scaling and saturation of output result.
- `NppStatus nppsMulC_32fc_I` (`Npp32fc` nValue, `Npp32fc` *pSrcDst, int nLength)
32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal times constant.
- `NppStatus nppsMulC_32fc` (const `Npp32fc` *pSrc, `Npp32fc` nValue, `Npp32fc` *pDst, int nLength)
32-bit floating point complex number (32 bit real, 32 bit imaginary) signal times constant.
- `NppStatus nppsMulC_64f_I` (`Npp64f` nValue, `Npp64f` *pSrcDst, int nLength)
64-bit floating point, in place signal times constant.
- `NppStatus nppsMulC_64f` (const `Npp64f` *pSrc, `Npp64f` nValue, `Npp64f` *pDst, int nLength)
64-bit floating point signal times constant.
- `NppStatus nppsMulC_64f64s_ISfs` (`Npp64f` nValue, `Npp64s` *pDst, int nLength, int nScaleFactor)
64-bit floating point signal times constant with in place conversion to 64-bit signed integer and with scaling and saturation of output result.
- `NppStatus nppsMulC_64fc_I` (`Npp64fc` nValue, `Npp64fc` *pSrcDst, int nLength)
64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal times constant.
- `NppStatus nppsMulC_64fc` (const `Npp64fc` *pSrc, `Npp64fc` nValue, `Npp64fc` *pDst, int nLength)
64-bit floating point complex number (64 bit real, 64 bit imaginary) signal times constant.

SubC

Subtracts a constant from each sample of a signal.

- `NppStatus nppsSubC_8u_ISfs` (`Npp8u` nValue, `Npp8u` *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char in place signal subtract constant, scale, then clamp to saturated value
- `NppStatus nppsSubC_8u_Sfs` (const `Npp8u` *pSrc, `Npp8u` nValue, `Npp8u` *pDst, int nLength, int nScaleFactor)
8-bit unsigned char signal subtract constant, scale, then clamp to saturated value.
- `NppStatus nppsSubC_16u_ISfs` (`Npp16u` nValue, `Npp16u` *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short in place signal subtract constant, scale, then clamp to saturated value.
- `NppStatus nppsSubC_16u_Sfs` (const `Npp16u` *pSrc, `Npp16u` nValue, `Npp16u` *pDst, int nLength, int nScaleFactor)
16-bit unsigned short signal subtract constant, scale, then clamp to saturated value.

- **NppStatus nppsSubC_16s_ISfs** (**Npp16s** nValue, **Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short in place signal subtract constant, scale, then clamp to saturated value.
- **NppStatus nppsSubC_16s_Sfs** (const **Npp16s** *pSrc, **Npp16s** nValue, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal subtract constant, scale, then clamp to saturated value.
- **NppStatus nppsSubC_16sc_ISfs** (**Npp16sc** nValue, **Npp16sc** *pSrcDst, int nLength, int nScaleFactor)
16-bit integer complex number (16 bit real, 16 bit imaginary) signal subtract constant, scale, then clamp to saturated value.
- **NppStatus nppsSubC_16sc_Sfs** (const **Npp16sc** *pSrc, **Npp16sc** nValue, **Npp16sc** *pDst, int nLength, int nScaleFactor)
16-bit integer complex number (16 bit real, 16 bit imaginary) signal subtract constant, scale, then clamp to saturated value.
- **NppStatus nppsSubC_32s_ISfs** (**Npp32s** nValue, **Npp32s** *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer in place signal subtract constant and scale.
- **NppStatus nppsSubC_32s_Sfs** (const **Npp32s** *pSrc, **Npp32s** nValue, **Npp32s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal subtract constant and scale.
- **NppStatus nppsSubC_32sc_ISfs** (**Npp32sc** nValue, **Npp32sc** *pSrcDst, int nLength, int nScaleFactor)
32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal subtract constant and scale.
- **NppStatus nppsSubC_32sc_Sfs** (const **Npp32sc** *pSrc, **Npp32sc** nValue, **Npp32sc** *pDst, int nLength, int nScaleFactor)
32-bit integer complex number (32 bit real, 32 bit imaginary) signal subtract constant and scale.
- **NppStatus nppsSubC_32f_I** (**Npp32f** nValue, **Npp32f** *pSrcDst, int nLength)
32-bit floating point in place signal subtract constant.
- **NppStatus nppsSubC_32f** (const **Npp32f** *pSrc, **Npp32f** nValue, **Npp32f** *pDst, int nLength)
32-bit floating point signal subtract constant.
- **NppStatus nppsSubC_32fc_I** (**Npp32fc** nValue, **Npp32fc** *pSrcDst, int nLength)
32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal subtract constant.
- **NppStatus nppsSubC_32fc** (const **Npp32fc** *pSrc, **Npp32fc** nValue, **Npp32fc** *pDst, int nLength)
32-bit floating point complex number (32 bit real, 32 bit imaginary) signal subtract constant.
- **NppStatus nppsSubC_64f_I** (**Npp64f** nValue, **Npp64f** *pSrcDst, int nLength)
64-bit floating point, in place signal subtract constant.
- **NppStatus nppsSubC_64f** (const **Npp64f** *pSrc, **Npp64f** nValue, **Npp64f** *pDst, int nLength)
64-bit floating point signal subtract constant.
- **NppStatus nppsSubC_64fc_I** (**Npp64fc** nValue, **Npp64fc** *pSrcDst, int nLength)

64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal subtract constant.

- **NppStatus nppsSubC_64fc** (const **Npp64fc** *pSrc, **Npp64fc** nValue, **Npp64fc** *pDst, int nLength)
64-bit floating point complex number (64 bit real, 64 bit imaginary) signal subtract constant.

SubCRev

Subtracts each sample of a signal from a constant.

- **NppStatus nppsSubCRev_8u_ISfs** (**Npp8u** nValue, **Npp8u** *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char in place signal subtract from constant, scale, then clamp to saturated value
- **NppStatus nppsSubCRev_8u_Sfs** (const **Npp8u** *pSrc, **Npp8u** nValue, **Npp8u** *pDst, int nLength, int nScaleFactor)
8-bit unsigned char signal subtract from constant, scale, then clamp to saturated value.
- **NppStatus nppsSubCRev_16u_ISfs** (**Npp16u** nValue, **Npp16u** *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short in place signal subtract from constant, scale, then clamp to saturated value.
- **NppStatus nppsSubCRev_16u_Sfs** (const **Npp16u** *pSrc, **Npp16u** nValue, **Npp16u** *pDst, int nLength, int nScaleFactor)
16-bit unsigned short signal subtract from constant, scale, then clamp to saturated value.
- **NppStatus nppsSubCRev_16s_ISfs** (**Npp16s** nValue, **Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short in place signal subtract from constant, scale, then clamp to saturated value.
- **NppStatus nppsSubCRev_16s_Sfs** (const **Npp16s** *pSrc, **Npp16s** nValue, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal subtract from constant, scale, then clamp to saturated value.
- **NppStatus nppsSubCRev_16sc_ISfs** (**Npp16sc** nValue, **Npp16sc** *pSrcDst, int nLength, int nScaleFactor)
16-bit integer complex number (16 bit real, 16 bit imaginary) signal subtract from constant, scale, then clamp to saturated value.
- **NppStatus nppsSubCRev_16sc_Sfs** (const **Npp16sc** *pSrc, **Npp16sc** nValue, **Npp16sc** *pDst, int nLength, int nScaleFactor)
16-bit integer complex number (16 bit real, 16 bit imaginary) signal subtract from constant, scale, then clamp to saturated value.
- **NppStatus nppsSubCRev_32s_ISfs** (**Npp32s** nValue, **Npp32s** *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer in place signal subtract from constant and scale.
- **NppStatus nppsSubCRev_32s_Sfs** (const **Npp32s** *pSrc, **Npp32s** nValue, **Npp32s** *pDst, int nLength, int nScaleFactor)
32-bit signed integersignal subtract from constant and scale.

- **NppStatus nppsSubCRev_32sc_ISfs** (**Npp32sc** nValue, **Npp32sc** *pSrcDst, int nLength, int nScaleFactor)
32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal subtract from constant and scale.
- **NppStatus nppsSubCRev_32sc_Sfs** (const **Npp32sc** *pSrc, **Npp32sc** nValue, **Npp32sc** *pDst, int nLength, int nScaleFactor)
32-bit integer complex number (32 bit real, 32 bit imaginary) signal subtract from constant and scale.
- **NppStatus nppsSubCRev_32f_I** (**Npp32f** nValue, **Npp32f** *pSrcDst, int nLength)
32-bit floating point in place signal subtract from constant.
- **NppStatus nppsSubCRev_32f** (const **Npp32f** *pSrc, **Npp32f** nValue, **Npp32f** *pDst, int nLength)
32-bit floating point signal subtract from constant.
- **NppStatus nppsSubCRev_32fc_I** (**Npp32fc** nValue, **Npp32fc** *pSrcDst, int nLength)
32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal subtract from constant.
- **NppStatus nppsSubCRev_32fc** (const **Npp32fc** *pSrc, **Npp32fc** nValue, **Npp32fc** *pDst, int nLength)
32-bit floating point complex number (32 bit real, 32 bit imaginary) signal subtract from constant.
- **NppStatus nppsSubCRev_64f_I** (**Npp64f** nValue, **Npp64f** *pSrcDst, int nLength)
64-bit floating point, in place signal subtract from constant.
- **NppStatus nppsSubCRev_64f** (const **Npp64f** *pSrc, **Npp64f** nValue, **Npp64f** *pDst, int nLength)
64-bit floating point signal subtract from constant.
- **NppStatus nppsSubCRev_64fc_I** (**Npp64fc** nValue, **Npp64fc** *pSrcDst, int nLength)
64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal subtract from constant.
- **NppStatus nppsSubCRev_64fc** (const **Npp64fc** *pSrc, **Npp64fc** nValue, **Npp64fc** *pDst, int nLength)
64-bit floating point complex number (64 bit real, 64 bit imaginary) signal subtract from constant.

DivC

Divides each sample of a signal by a constant.

- **NppStatus nppsDivC_8u_ISfs** (**Npp8u** nValue, **Npp8u** *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char in place signal divided by constant, scale, then clamp to saturated value
- **NppStatus nppsDivC_8u_Sfs** (const **Npp8u** *pSrc, **Npp8u** nValue, **Npp8u** *pDst, int nLength, int nScaleFactor)
8-bit unsigned char signal divided by constant, scale, then clamp to saturated value.
- **NppStatus nppsDivC_16u_ISfs** (**Npp16u** nValue, **Npp16u** *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short in place signal divided by constant, scale, then clamp to saturated value.

- **NppStatus nppsDivC_16u_Sfs** (const **Npp16u** *pSrc, **Npp16u** nValue, **Npp16u** *pDst, int nLength, int nScaleFactor)
16-bit unsigned short signal divided by constant, scale, then clamp to saturated value.
- **NppStatus nppsDivC_16s_ISfs** (**Npp16s** nValue, **Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short in place signal divided by constant, scale, then clamp to saturated value.
- **NppStatus nppsDivC_16s_Sfs** (const **Npp16s** *pSrc, **Npp16s** nValue, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal divided by constant, scale, then clamp to saturated value.
- **NppStatus nppsDivC_16sc_ISfs** (**Npp16sc** nValue, **Npp16sc** *pSrcDst, int nLength, int nScaleFactor)
16-bit integer complex number (16 bit real, 16 bit imaginary) signal divided by constant, scale, then clamp to saturated value.
- **NppStatus nppsDivC_16sc_Sfs** (const **Npp16sc** *pSrc, **Npp16sc** nValue, **Npp16sc** *pDst, int nLength, int nScaleFactor)
16-bit integer complex number (16 bit real, 16 bit imaginary) signal divided by constant, scale, then clamp to saturated value.
- **NppStatus nppsDivC_32s_ISfs** (**Npp32s** nValue, **Npp32s** *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer in place signal divided by constant and scale.
- **NppStatus nppsDivC_32s_Sfs** (const **Npp32s** *pSrc, **Npp32s** nValue, **Npp32s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal divided by constant and scale.
- **NppStatus nppsDivC_32sc_ISfs** (**Npp32sc** nValue, **Npp32sc** *pSrcDst, int nLength, int nScaleFactor)
32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal divided by constant and scale.
- **NppStatus nppsDivC_32sc_Sfs** (const **Npp32sc** *pSrc, **Npp32sc** nValue, **Npp32sc** *pDst, int nLength, int nScaleFactor)
32-bit integer complex number (32 bit real, 32 bit imaginary) signal divided by constant and scale.
- **NppStatus nppsDivC_32f_I** (**Npp32f** nValue, **Npp32f** *pSrcDst, int nLength)
32-bit floating point in place signal divided by constant.
- **NppStatus nppsDivC_32f** (const **Npp32f** *pSrc, **Npp32f** nValue, **Npp32f** *pDst, int nLength)
32-bit floating point signal divided by constant.
- **NppStatus nppsDivC_32fc_I** (**Npp32fc** nValue, **Npp32fc** *pSrcDst, int nLength)
32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal divided by constant.
- **NppStatus nppsDivC_32fc** (const **Npp32fc** *pSrc, **Npp32fc** nValue, **Npp32fc** *pDst, int nLength)
32-bit floating point complex number (32 bit real, 32 bit imaginary) signal divided by constant.
- **NppStatus nppsDivC_64f_I** (**Npp64f** nValue, **Npp64f** *pSrcDst, int nLength)
64-bit floating point in place signal divided by constant.

- `NppStatus nppsDivC_64f` (const `Npp64f` *pSrc, `Npp64f` nValue, `Npp64f` *pDst, int nLength)
64-bit floating point signal divided by constant.
- `NppStatus nppsDivC_64fc_I` (`Npp64fc` nValue, `Npp64fc` *pSrcDst, int nLength)
64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal divided by constant.
- `NppStatus nppsDivC_64fc` (const `Npp64fc` *pSrc, `Npp64fc` nValue, `Npp64fc` *pDst, int nLength)
64-bit floating point complex number (64 bit real, 64 bit imaginary) signal divided by constant.

DivCRev

Divides a constant by each sample of a signal.

- `NppStatus nppsDivCRev_8u_I` (`Npp8u` nValue, `Npp8u` *pSrcDst, int nLength)
8-bit unsigned char signal in place constant divided by signal, scale, then clamp to saturated value
- `NppStatus nppsDivCRev_8u` (const `Npp8u` *pSrc, `Npp8u` nValue, `Npp8u` *pDst, int nLength)
8-bit unsigned char signal divided by constant, then clamp to saturated value.
- `NppStatus nppsDivCRev_16u_I` (`Npp16u` nValue, `Npp16u` *pSrcDst, int nLength)
16-bit unsigned short in place constant divided by signal, then clamp to saturated value.
- `NppStatus nppsDivCRev_16u` (const `Npp16u` *pSrc, `Npp16u` nValue, `Npp16u` *pDst, int nLength)
16-bit unsigned short signal divided by constant, then clamp to saturated value.
- `NppStatus nppsDivCRev_16s_I` (`Npp16s` nValue, `Npp16s` *pSrcDst, int nLength)
16-bit signed short in place constant divided by signal, then clamp to saturated value.
- `NppStatus nppsDivCRev_16s` (const `Npp16s` *pSrc, `Npp16s` nValue, `Npp16s` *pDst, int nLength)
16-bit signed short constant divided by signal, then clamp to saturated value.
- `NppStatus nppsDivCRev_32s_I` (`Npp32s` nValue, `Npp32s` *pSrcDst, int nLength)
32-bit signed integer in place constant divided by signal.
- `NppStatus nppsDivCRev_32s` (const `Npp32s` *pSrc, `Npp32s` nValue, `Npp32s` *pDst, int nLength)
32-bit signed integer constant divided by signal.
- `NppStatus nppsDivCRev_32f_I` (`Npp32f` nValue, `Npp32f` *pSrcDst, int nLength)
32-bit floating point in place constant divided by signal.
- `NppStatus nppsDivCRev_32f` (const `Npp32f` *pSrc, `Npp32f` nValue, `Npp32f` *pDst, int nLength)
32-bit floating point constant divided by signal.
- `NppStatus nppsDivCRev_64f_I` (`Npp64f` nValue, `Npp64f` *pSrcDst, int nLength)
64-bit floating point in place constant divided by signal.
- `NppStatus nppsDivCRev_64f` (const `Npp64f` *pSrc, `Npp64f` nValue, `Npp64f` *pDst, int nLength)
64-bit floating point constant divided by signal.

Add Signal

Sample by sample addition of two signals.

- **NppStatus** **nppsAdd_16s** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp16s** *pDst, int nLength)
16-bit signed short signal add signal, then clamp to saturated value.
- **NppStatus** **nppsAdd_16u** (const **Npp16u** *pSrc1, const **Npp16u** *pSrc2, **Npp16u** *pDst, int nLength)
16-bit unsigned short signal add signal, then clamp to saturated value.
- **NppStatus** **nppsAdd_32u** (const **Npp32u** *pSrc1, const **Npp32u** *pSrc2, **Npp32u** *pDst, int nLength)
32-bit unsigned int signal add signal, then clamp to saturated value.
- **NppStatus** **nppsAdd_32f** (const **Npp32f** *pSrc1, const **Npp32f** *pSrc2, **Npp32f** *pDst, int nLength)
32-bit floating point signal add signal, then clamp to saturated value.
- **NppStatus** **nppsAdd_64f** (const **Npp64f** *pSrc1, const **Npp64f** *pSrc2, **Npp64f** *pDst, int nLength)
64-bit floating point signal add signal, then clamp to saturated value.
- **NppStatus** **nppsAdd_32fc** (const **Npp32fc** *pSrc1, const **Npp32fc** *pSrc2, **Npp32fc** *pDst, int nLength)
32-bit complex floating point signal add signal, then clamp to saturated value.
- **NppStatus** **nppsAdd_64fc** (const **Npp64fc** *pSrc1, const **Npp64fc** *pSrc2, **Npp64fc** *pDst, int nLength)
64-bit complex floating point signal add signal, then clamp to saturated value.
- **NppStatus** **nppsAdd_8u16u** (const **Npp8u** *pSrc1, const **Npp8u** *pSrc2, **Npp16u** *pDst, int nLength)
8-bit unsigned char signal add signal with 16-bit unsigned result, then clamp to saturated value.
- **NppStatus** **nppsAdd_16s32f** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp32f** *pDst, int nLength)
16-bit signed short signal add signal with 32-bit floating point result, then clamp to saturated value.
- **NppStatus** **nppsAdd_8u_Sfs** (const **Npp8u** *pSrc1, const **Npp8u** *pSrc2, **Npp8u** *pDst, int nLength, int nScaleFactor)
8-bit unsigned char add signal, scale, then clamp to saturated value.
- **NppStatus** **nppsAdd_16u_Sfs** (const **Npp16u** *pSrc1, const **Npp16u** *pSrc2, **Npp16u** *pDst, int nLength, int nScaleFactor)
16-bit unsigned short add signal, scale, then clamp to saturated value.
- **NppStatus** **nppsAdd_16s_Sfs** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short add signal, scale, then clamp to saturated value.

- `NppStatus nppsAdd_32s_Sfs` (const `Npp32s` *pSrc1, const `Npp32s` *pSrc2, `Npp32s` *pDst, int nLength, int nScaleFactor)
32-bit signed integer add signal, scale, then clamp to saturated value.
- `NppStatus nppsAdd_64s_Sfs` (const `Npp64s` *pSrc1, const `Npp64s` *pSrc2, `Npp64s` *pDst, int nLength, int nScaleFactor)
64-bit signed integer add signal, scale, then clamp to saturated value.
- `NppStatus nppsAdd_16sc_Sfs` (const `Npp16sc` *pSrc1, const `Npp16sc` *pSrc2, `Npp16sc` *pDst, int nLength, int nScaleFactor)
16-bit signed complex short add signal, scale, then clamp to saturated value.
- `NppStatus nppsAdd_32sc_Sfs` (const `Npp32sc` *pSrc1, const `Npp32sc` *pSrc2, `Npp32sc` *pDst, int nLength, int nScaleFactor)
32-bit signed complex integer add signal, scale, then clamp to saturated value.
- `NppStatus nppsAdd_16s_I` (const `Npp16s` *pSrc, `Npp16s` *pSrcDst, int nLength)
16-bit signed short in place signal add signal, then clamp to saturated value.
- `NppStatus nppsAdd_32f_I` (const `Npp32f` *pSrc, `Npp32f` *pSrcDst, int nLength)
32-bit floating point in place signal add signal, then clamp to saturated value.
- `NppStatus nppsAdd_64f_I` (const `Npp64f` *pSrc, `Npp64f` *pSrcDst, int nLength)
64-bit floating point in place signal add signal, then clamp to saturated value.
- `NppStatus nppsAdd_32fc_I` (const `Npp32fc` *pSrc, `Npp32fc` *pSrcDst, int nLength)
32-bit complex floating point in place signal add signal, then clamp to saturated value.
- `NppStatus nppsAdd_64fc_I` (const `Npp64fc` *pSrc, `Npp64fc` *pSrcDst, int nLength)
64-bit complex floating point in place signal add signal, then clamp to saturated value.
- `NppStatus nppsAdd_16s32s_I` (const `Npp16s` *pSrc, `Npp32s` *pSrcDst, int nLength)
16/32-bit signed short in place signal add signal with 32-bit signed integer results, then clamp to saturated value.
- `NppStatus nppsAdd_8u_ISfs` (const `Npp8u` *pSrc, `Npp8u` *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char in place signal add signal, with scaling, then clamp to saturated value.
- `NppStatus nppsAdd_16u_ISfs` (const `Npp16u` *pSrc, `Npp16u` *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short in place signal add signal, with scaling, then clamp to saturated value.
- `NppStatus nppsAdd_16s_ISfs` (const `Npp16s` *pSrc, `Npp16s` *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short in place signal add signal, with scaling, then clamp to saturated value.
- `NppStatus nppsAdd_32s_ISfs` (const `Npp32s` *pSrc, `Npp32s` *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer in place signal add signal, with scaling, then clamp to saturated value.

- **NppStatus** **nppsAdd_16sc_ISfs** (const **Npp16sc** *pSrc, **Npp16sc** *pSrcDst, int nLength, int nScaleFactor)

16-bit complex signed short in place signal add signal, with scaling, then clamp to saturated value.

- **NppStatus** **nppsAdd_32sc_ISfs** (const **Npp32sc** *pSrc, **Npp32sc** *pSrcDst, int nLength, int nScaleFactor)

32-bit complex signed integer in place signal add signal, with scaling, then clamp to saturated value.

AddProduct Signal

Adds sample by sample product of two signals to the destination signal.

- **NppStatus** **nppsAddProduct_32f** (const **Npp32f** *pSrc1, const **Npp32f** *pSrc2, **Npp32f** *pDst, int nLength)

32-bit floating point signal add product of source signal times destination signal to destination signal, then clamp to saturated value.

- **NppStatus** **nppsAddProduct_64f** (const **Npp64f** *pSrc1, const **Npp64f** *pSrc2, **Npp64f** *pDst, int nLength)

64-bit floating point signal add product of source signal times destination signal to destination signal, then clamp to saturated value.

- **NppStatus** **nppsAddProduct_32fc** (const **Npp32fc** *pSrc1, const **Npp32fc** *pSrc2, **Npp32fc** *pDst, int nLength)

32-bit complex floating point signal add product of source signal times destination signal to destination signal, then clamp to saturated value.

- **NppStatus** **nppsAddProduct_64fc** (const **Npp64fc** *pSrc1, const **Npp64fc** *pSrc2, **Npp64fc** *pDst, int nLength)

64-bit complex floating point signal add product of source signal times destination signal to destination signal, then clamp to saturated value.

- **NppStatus** **nppsAddProduct_16s_Sfs** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp16s** *pDst, int nLength, int nScaleFactor)

16-bit signed short signal add product of source signal1 times source signal2 to destination signal, with scaling, then clamp to saturated value.

- **NppStatus** **nppsAddProduct_32s_Sfs** (const **Npp32s** *pSrc1, const **Npp32s** *pSrc2, **Npp32s** *pDst, int nLength, int nScaleFactor)

32-bit signed short signal add product of source signal1 times source signal2 to destination signal, with scaling, then clamp to saturated value.

- **NppStatus** **nppsAddProduct_16s32s_Sfs** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp32s** *pDst, int nLength, int nScaleFactor)

16-bit signed short signal add product of source signal1 times source signal2 to 32-bit signed integer destination signal, with scaling, then clamp to saturated value.

Mul Signal

Sample by sample multiplication the samples of two signals.

- **NppStatus nppsMul_16s** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp16s** *pDst, int nLength)
16-bit signed short signal times signal, then clamp to saturated value.
- **NppStatus nppsMul_32f** (const **Npp32f** *pSrc1, const **Npp32f** *pSrc2, **Npp32f** *pDst, int nLength)
32-bit floating point signal times signal, then clamp to saturated value.
- **NppStatus nppsMul_64f** (const **Npp64f** *pSrc1, const **Npp64f** *pSrc2, **Npp64f** *pDst, int nLength)
64-bit floating point signal times signal, then clamp to saturated value.
- **NppStatus nppsMul_32fc** (const **Npp32fc** *pSrc1, const **Npp32fc** *pSrc2, **Npp32fc** *pDst, int nLength)
32-bit complex floating point signal times signal, then clamp to saturated value.
- **NppStatus nppsMul_64fc** (const **Npp64fc** *pSrc1, const **Npp64fc** *pSrc2, **Npp64fc** *pDst, int nLength)
64-bit complex floating point signal times signal, then clamp to saturated value.
- **NppStatus nppsMul_8u16u** (const **Npp8u** *pSrc1, const **Npp8u** *pSrc2, **Npp16u** *pDst, int nLength)
8-bit unsigned char signal times signal with 16-bit unsigned result, then clamp to saturated value.
- **NppStatus nppsMul_16s32f** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp32f** *pDst, int nLength)
16-bit signed short signal times signal with 32-bit floating point result, then clamp to saturated value.
- **NppStatus nppsMul_32f32fc** (const **Npp32f** *pSrc1, const **Npp32fc** *pSrc2, **Npp32fc** *pDst, int nLength)
32-bit floating point signal times 32-bit complex floating point signal with complex 32-bit floating point result, then clamp to saturated value.
- **NppStatus nppsMul_8u_Sfs** (const **Npp8u** *pSrc1, const **Npp8u** *pSrc2, **Npp8u** *pDst, int nLength, int nScaleFactor)
8-bit unsigned char signal times signal, scale, then clamp to saturated value.
- **NppStatus nppsMul_16u_Sfs** (const **Npp16u** *pSrc1, const **Npp16u** *pSrc2, **Npp16u** *pDst, int nLength, int nScaleFactor)
16-bit unsigned short signal time signal, scale, then clamp to saturated value.
- **NppStatus nppsMul_16s_Sfs** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal times signal, scale, then clamp to saturated value.
- **NppStatus nppsMul_32s_Sfs** (const **Npp32s** *pSrc1, const **Npp32s** *pSrc2, **Npp32s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal times signal, scale, then clamp to saturated value.

- **NppStatus nppsMul_16sc_Sfs** (const **Npp16sc** *pSrc1, const **Npp16sc** *pSrc2, **Npp16sc** *pDst, int nLength, int nScaleFactor)
16-bit signed complex short signal times signal, scale, then clamp to saturated value.
- **NppStatus nppsMul_32sc_Sfs** (const **Npp32sc** *pSrc1, const **Npp32sc** *pSrc2, **Npp32sc** *pDst, int nLength, int nScaleFactor)
32-bit signed complex integer signal times signal, scale, then clamp to saturated value.
- **NppStatus nppsMul_16u16s_Sfs** (const **Npp16u** *pSrc1, const **Npp16s** *pSrc2, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit unsigned short signal times 16-bit signed short signal, scale, then clamp to 16-bit signed saturated value.
- **NppStatus nppsMul_16s32s_Sfs** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp32s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal times signal, scale, then clamp to 32-bit signed saturated value.
- **NppStatus nppsMul_32s32sc_Sfs** (const **Npp32s** *pSrc1, const **Npp32sc** *pSrc2, **Npp32sc** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal times 32-bit complex signed integer signal, scale, then clamp to 32-bit complex integer saturated value.
- **NppStatus nppsMul_Low_32s_Sfs** (const **Npp32s** *pSrc1, const **Npp32s** *pSrc2, **Npp32s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal times signal, scale, then clamp to saturated value.
- **NppStatus nppsMul_16s_I** (const **Npp16s** *pSrc, **Npp16s** *pSrcDst, int nLength)
16-bit signed short in place signal times signal, then clamp to saturated value.
- **NppStatus nppsMul_32f_I** (const **Npp32f** *pSrc, **Npp32f** *pSrcDst, int nLength)
32-bit floating point in place signal times signal, then clamp to saturated value.
- **NppStatus nppsMul_64f_I** (const **Npp64f** *pSrc, **Npp64f** *pSrcDst, int nLength)
64-bit floating point in place signal times signal, then clamp to saturated value.
- **NppStatus nppsMul_32fc_I** (const **Npp32fc** *pSrc, **Npp32fc** *pSrcDst, int nLength)
32-bit complex floating point in place signal times signal, then clamp to saturated value.
- **NppStatus nppsMul_64fc_I** (const **Npp64fc** *pSrc, **Npp64fc** *pSrcDst, int nLength)
64-bit complex floating point in place signal times signal, then clamp to saturated value.
- **NppStatus nppsMul_32f32fc_I** (const **Npp32f** *pSrc, **Npp32fc** *pSrcDst, int nLength)
32-bit complex floating point in place signal times 32-bit floating point signal, then clamp to 32-bit complex floating point saturated value.
- **NppStatus nppsMul_8u_ISfs** (const **Npp8u** *pSrc, **Npp8u** *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char in place signal times signal, with scaling, then clamp to saturated value.
- **NppStatus nppsMul_16u_ISfs** (const **Npp16u** *pSrc, **Npp16u** *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short in place signal times signal, with scaling, then clamp to saturated value.

- **NppStatus nppsMul_16s_ISfs** (const **Npp16s** *pSrc, **Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short in place signal times signal, with scaling, then clamp to saturated value.
- **NppStatus nppsMul_32s_ISfs** (const **Npp32s** *pSrc, **Npp32s** *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer in place signal times signal, with scaling, then clamp to saturated value.
- **NppStatus nppsMul_16sc_ISfs** (const **Npp16sc** *pSrc, **Npp16sc** *pSrcDst, int nLength, int nScaleFactor)
16-bit complex signed short in place signal times signal, with scaling, then clamp to saturated value.
- **NppStatus nppsMul_32sc_ISfs** (const **Npp32sc** *pSrc, **Npp32sc** *pSrcDst, int nLength, int nScaleFactor)
32-bit complex signed integer in place signal times signal, with scaling, then clamp to saturated value.
- **NppStatus nppsMul_32s32sc_ISfs** (const **Npp32s** *pSrc, **Npp32sc** *pSrcDst, int nLength, int nScaleFactor)
32-bit complex signed integer in place signal times 32-bit signed integer signal, with scaling, then clamp to saturated value.

Sub Signal

Sample by sample subtraction of the samples of two signals.

- **NppStatus nppsSub_16s** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp16s** *pDst, int nLength)
16-bit signed short signal subtract signal, then clamp to saturated value.
- **NppStatus nppsSub_32f** (const **Npp32f** *pSrc1, const **Npp32f** *pSrc2, **Npp32f** *pDst, int nLength)
32-bit floating point signal subtract signal, then clamp to saturated value.
- **NppStatus nppsSub_64f** (const **Npp64f** *pSrc1, const **Npp64f** *pSrc2, **Npp64f** *pDst, int nLength)
64-bit floating point signal subtract signal, then clamp to saturated value.
- **NppStatus nppsSub_32fc** (const **Npp32fc** *pSrc1, const **Npp32fc** *pSrc2, **Npp32fc** *pDst, int nLength)
32-bit complex floating point signal subtract signal, then clamp to saturated value.
- **NppStatus nppsSub_64fc** (const **Npp64fc** *pSrc1, const **Npp64fc** *pSrc2, **Npp64fc** *pDst, int nLength)
64-bit complex floating point signal subtract signal, then clamp to saturated value.
- **NppStatus nppsSub_16s32f** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp32f** *pDst, int nLength)
16-bit signed short signal subtract 16-bit signed short signal, then clamp and convert to 32-bit floating point saturated value.
- **NppStatus nppsSub_8u_Sfs** (const **Npp8u** *pSrc1, const **Npp8u** *pSrc2, **Npp8u** *pDst, int nLength, int nScaleFactor)

8-bit unsigned char signal subtract signal, scale, then clamp to saturated value.

- `NppStatus nppsSub_16u_Sfs` (const `Npp16u` *pSrc1, const `Npp16u` *pSrc2, `Npp16u` *pDst, int nLength, int nScaleFactor)

16-bit unsigned short signal subtract signal, scale, then clamp to saturated value.

- `NppStatus nppsSub_16s_Sfs` (const `Npp16s` *pSrc1, const `Npp16s` *pSrc2, `Npp16s` *pDst, int nLength, int nScaleFactor)

16-bit signed short signal subtract signal, scale, then clamp to saturated value.

- `NppStatus nppsSub_32s_Sfs` (const `Npp32s` *pSrc1, const `Npp32s` *pSrc2, `Npp32s` *pDst, int nLength, int nScaleFactor)

32-bit signed integer signal subtract signal, scale, then clamp to saturated value.

- `NppStatus nppsSub_16sc_Sfs` (const `Npp16sc` *pSrc1, const `Npp16sc` *pSrc2, `Npp16sc` *pDst, int nLength, int nScaleFactor)

16-bit signed complex short signal subtract signal, scale, then clamp to saturated value.

- `NppStatus nppsSub_32sc_Sfs` (const `Npp32sc` *pSrc1, const `Npp32sc` *pSrc2, `Npp32sc` *pDst, int nLength, int nScaleFactor)

32-bit signed complex integer signal subtract signal, scale, then clamp to saturated value.

- `NppStatus nppsSub_16s_I` (const `Npp16s` *pSrc, `Npp16s` *pSrcDst, int nLength)

16-bit signed short in place signal subtract signal, then clamp to saturated value.

- `NppStatus nppsSub_32f_I` (const `Npp32f` *pSrc, `Npp32f` *pSrcDst, int nLength)

32-bit floating point in place signal subtract signal, then clamp to saturated value.

- `NppStatus nppsSub_64f_I` (const `Npp64f` *pSrc, `Npp64f` *pSrcDst, int nLength)

64-bit floating point in place signal subtract signal, then clamp to saturated value.

- `NppStatus nppsSub_32fc_I` (const `Npp32fc` *pSrc, `Npp32fc` *pSrcDst, int nLength)

32-bit complex floating point in place signal subtract signal, then clamp to saturated value.

- `NppStatus nppsSub_64fc_I` (const `Npp64fc` *pSrc, `Npp64fc` *pSrcDst, int nLength)

64-bit complex floating point in place signal subtract signal, then clamp to saturated value.

- `NppStatus nppsSub_8u_ISfs` (const `Npp8u` *pSrc, `Npp8u` *pSrcDst, int nLength, int nScaleFactor)

8-bit unsigned char in place signal subtract signal, with scaling, then clamp to saturated value.

- `NppStatus nppsSub_16u_ISfs` (const `Npp16u` *pSrc, `Npp16u` *pSrcDst, int nLength, int nScaleFactor)

16-bit unsigned short in place signal subtract signal, with scaling, then clamp to saturated value.

- `NppStatus nppsSub_16s_ISfs` (const `Npp16s` *pSrc, `Npp16s` *pSrcDst, int nLength, int nScaleFactor)

16-bit signed short in place signal subtract signal, with scaling, then clamp to saturated value.

- `NppStatus nppsSub_32s_ISfs` (const `Npp32s` *pSrc, `Npp32s` *pSrcDst, int nLength, int nScaleFactor)

32-bit signed integer in place signal subtract signal, with scaling, then clamp to saturated value.

- **NppStatus** **nppsSub_16sc_ISfs** (const **Npp16sc** *pSrc, **Npp16sc** *pSrcDst, int nLength, int nScaleFactor)

16-bit complex signed short in place signal subtract signal, with scaling, then clamp to saturated value.

- **NppStatus** **nppsSub_32sc_ISfs** (const **Npp32sc** *pSrc, **Npp32sc** *pSrcDst, int nLength, int nScaleFactor)

32-bit complex signed integer in place signal subtract signal, with scaling, then clamp to saturated value.

Div Signal

Sample by sample division of the samples of two signals.

- **NppStatus** **nppsDiv_8u_Sfs** (const **Npp8u** *pSrc1, const **Npp8u** *pSrc2, **Npp8u** *pDst, int nLength, int nScaleFactor)

8-bit unsigned char signal divide signal, scale, then clamp to saturated value.

- **NppStatus** **nppsDiv_16u_Sfs** (const **Npp16u** *pSrc1, const **Npp16u** *pSrc2, **Npp16u** *pDst, int nLength, int nScaleFactor)

16-bit unsigned short signal divide signal, scale, then clamp to saturated value.

- **NppStatus** **nppsDiv_16s_Sfs** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp16s** *pDst, int nLength, int nScaleFactor)

16-bit signed short signal divide signal, scale, then clamp to saturated value.

- **NppStatus** **nppsDiv_32s_Sfs** (const **Npp32s** *pSrc1, const **Npp32s** *pSrc2, **Npp32s** *pDst, int nLength, int nScaleFactor)

32-bit signed integer signal divide signal, scale, then clamp to saturated value.

- **NppStatus** **nppsDiv_16sc_Sfs** (const **Npp16sc** *pSrc1, const **Npp16sc** *pSrc2, **Npp16sc** *pDst, int nLength, int nScaleFactor)

16-bit signed complex short signal divide signal, scale, then clamp to saturated value.

- **NppStatus** **nppsDiv_32s16s_Sfs** (const **Npp16s** *pSrc1, const **Npp32s** *pSrc2, **Npp16s** *pDst, int nLength, int nScaleFactor)

32-bit signed integer signal divided by 16-bit signed short signal, scale, then clamp to 16-bit signed short saturated value.

- **NppStatus** **nppsDiv_32f** (const **Npp32f** *pSrc1, const **Npp32f** *pSrc2, **Npp32f** *pDst, int nLength)

32-bit floating point signal divide signal, then clamp to saturated value.

- **NppStatus** **nppsDiv_64f** (const **Npp64f** *pSrc1, const **Npp64f** *pSrc2, **Npp64f** *pDst, int nLength)

64-bit floating point signal divide signal, then clamp to saturated value.

- **NppStatus** **nppsDiv_32fc** (const **Npp32fc** *pSrc1, const **Npp32fc** *pSrc2, **Npp32fc** *pDst, int nLength)

32-bit complex floating point signal divide signal, then clamp to saturated value.

- **NppStatus nppsDiv_64fc** (const **Npp64fc** *pSrc1, const **Npp64fc** *pSrc2, **Npp64fc** *pDst, int nLength)
64-bit complex floating point signal divide signal, then clamp to saturated value.
- **NppStatus nppsDiv_8u_ISfs** (const **Npp8u** *pSrc, **Npp8u** *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char in place signal divide signal, with scaling, then clamp to saturated value.
- **NppStatus nppsDiv_16u_ISfs** (const **Npp16u** *pSrc, **Npp16u** *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short in place signal divide signal, with scaling, then clamp to saturated value.
- **NppStatus nppsDiv_16s_ISfs** (const **Npp16s** *pSrc, **Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short in place signal divide signal, with scaling, then clamp to saturated value.
- **NppStatus nppsDiv_16sc_ISfs** (const **Npp16sc** *pSrc, **Npp16sc** *pSrcDst, int nLength, int nScaleFactor)
16-bit complex signed short in place signal divide signal, with scaling, then clamp to saturated value.
- **NppStatus nppsDiv_32s_ISfs** (const **Npp32s** *pSrc, **Npp32s** *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer in place signal divide signal, with scaling, then clamp to saturated value.
- **NppStatus nppsDiv_32f_I** (const **Npp32f** *pSrc, **Npp32f** *pSrcDst, int nLength)
32-bit floating point in place signal divide signal, then clamp to saturated value.
- **NppStatus nppsDiv_64f_I** (const **Npp64f** *pSrc, **Npp64f** *pSrcDst, int nLength)
64-bit floating point in place signal divide signal, then clamp to saturated value.
- **NppStatus nppsDiv_32fc_I** (const **Npp32fc** *pSrc, **Npp32fc** *pSrcDst, int nLength)
32-bit complex floating point in place signal divide signal, then clamp to saturated value.
- **NppStatus nppsDiv_64fc_I** (const **Npp64fc** *pSrc, **Npp64fc** *pSrcDst, int nLength)
64-bit complex floating point in place signal divide signal, then clamp to saturated value.

Div_Round Signal

Sample by sample division of the samples of two signals with rounding.

- **NppStatus nppsDiv_Round_8u_Sfs** (const **Npp8u** *pSrc1, const **Npp8u** *pSrc2, **Npp8u** *pDst, int nLength, **NppRoundMode** nRndMode, int nScaleFactor)
8-bit unsigned char signal divide signal, scale, then clamp to saturated value.
- **NppStatus nppsDiv_Round_16u_Sfs** (const **Npp16u** *pSrc1, const **Npp16u** *pSrc2, **Npp16u** *pDst, int nLength, **NppRoundMode** nRndMode, int nScaleFactor)
16-bit unsigned short signal divide signal, scale, round, then clamp to saturated value.
- **NppStatus nppsDiv_Round_16s_Sfs** (const **Npp16s** *pSrc1, const **Npp16s** *pSrc2, **Npp16s** *pDst, int nLength, **NppRoundMode** nRndMode, int nScaleFactor)

16-bit signed short signal divide signal, scale, round, then clamp to saturated value.

- `NppStatus nppsDiv_Round_8u_ISfs` (const `Npp8u` *pSrc, `Npp8u` *pSrcDst, int nLength, `NppRoundMode` nRndMode, int nScaleFactor)

8-bit unsigned char in place signal divide signal, with scaling, rounding then clamp to saturated value.

- `NppStatus nppsDiv_Round_16u_ISfs` (const `Npp16u` *pSrc, `Npp16u` *pSrcDst, int nLength, `NppRoundMode` nRndMode, int nScaleFactor)

16-bit unsigned short in place signal divide signal, with scaling, rounding then clamp to saturated value.

- `NppStatus nppsDiv_Round_16s_ISfs` (const `Npp16s` *pSrc, `Npp16s` *pSrcDst, int nLength, `NppRoundMode` nRndMode, int nScaleFactor)

16-bit signed short in place signal divide signal, with scaling, rounding then clamp to saturated value.

Absolute Value Signal

Absolute value of each sample of a signal.

- `NppStatus nppsAbs_16s` (const `Npp16s` *pSrc, `Npp16s` *pDst, int nLength)

16-bit signed short signal absolute value.

- `NppStatus nppsAbs_32s` (const `Npp32s` *pSrc, `Npp32s` *pDst, int nLength)

32-bit signed integer signal absolute value.

- `NppStatus nppsAbs_32f` (const `Npp32f` *pSrc, `Npp32f` *pDst, int nLength)

32-bit floating point signal absolute value.

- `NppStatus nppsAbs_64f` (const `Npp64f` *pSrc, `Npp64f` *pDst, int nLength)

64-bit floating point signal absolute value.

- `NppStatus nppsAbs_16s_I` (`Npp16s` *pSrcDst, int nLength)

16-bit signed short signal absolute value.

- `NppStatus nppsAbs_32s_I` (`Npp32s` *pSrcDst, int nLength)

32-bit signed integer signal absolute value.

- `NppStatus nppsAbs_32f_I` (`Npp32f` *pSrcDst, int nLength)

32-bit floating point signal absolute value.

- `NppStatus nppsAbs_64f_I` (`Npp64f` *pSrcDst, int nLength)

64-bit floating point signal absolute value.

Square Signal

Squares each sample of a signal.

- `NppStatus nppsSqr_32f` (const `Npp32f` *pSrc, `Npp32f` *pDst, int nLength)

32-bit floating point signal squared.

- **NppStatus nppsSqr_64f** (const **Npp64f** *pSrc, **Npp64f** *pDst, int nLength)
64-bit floating point signal squared.
- **NppStatus nppsSqr_32fc** (const **Npp32fc** *pSrc, **Npp32fc** *pDst, int nLength)
32-bit complex floating point signal squared.
- **NppStatus nppsSqr_64fc** (const **Npp64fc** *pSrc, **Npp64fc** *pDst, int nLength)
64-bit complex floating point signal squared.
- **NppStatus nppsSqr_32f_I** (**Npp32f** *pSrcDst, int nLength)
32-bit floating point signal squared.
- **NppStatus nppsSqr_64f_I** (**Npp64f** *pSrcDst, int nLength)
64-bit floating point signal squared.
- **NppStatus nppsSqr_32fc_I** (**Npp32fc** *pSrcDst, int nLength)
32-bit complex floating point signal squared.
- **NppStatus nppsSqr_64fc_I** (**Npp64fc** *pSrcDst, int nLength)
64-bit complex floating point signal squared.
- **NppStatus nppsSqr_8u_Sfs** (const **Npp8u** *pSrc, **Npp8u** *pDst, int nLength, int nScaleFactor)
8-bit unsigned char signal squared, scale, then clamp to saturated value.
- **NppStatus nppsSqr_16u_Sfs** (const **Npp16u** *pSrc, **Npp16u** *pDst, int nLength, int nScaleFactor)
16-bit unsigned short signal squared, scale, then clamp to saturated value.
- **NppStatus nppsSqr_16s_Sfs** (const **Npp16s** *pSrc, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal squared, scale, then clamp to saturated value.
- **NppStatus nppsSqr_16sc_Sfs** (const **Npp16sc** *pSrc, **Npp16sc** *pDst, int nLength, int nScaleFactor)
16-bit complex signed short signal squared, scale, then clamp to saturated value.
- **NppStatus nppsSqr_8u_ISfs** (**Npp8u** *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char signal squared, scale, then clamp to saturated value.
- **NppStatus nppsSqr_16u_ISfs** (**Npp16u** *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short signal squared, scale, then clamp to saturated value.
- **NppStatus nppsSqr_16s_ISfs** (**Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short signal squared, scale, then clamp to saturated value.
- **NppStatus nppsSqr_16sc_ISfs** (**Npp16sc** *pSrcDst, int nLength, int nScaleFactor)
16-bit complex signed short signal squared, scale, then clamp to saturated value.

Square Root Signal

Square root of each sample of a signal.

- **NppStatus nppsSqrt_32f** (const **Npp32f** *pSrc, **Npp32f** *pDst, int nLength)
32-bit floating point signal square root.
- **NppStatus nppsSqrt_64f** (const **Npp64f** *pSrc, **Npp64f** *pDst, int nLength)
64-bit floating point signal square root.
- **NppStatus nppsSqrt_32fc** (const **Npp32fc** *pSrc, **Npp32fc** *pDst, int nLength)
32-bit complex floating point signal square root.
- **NppStatus nppsSqrt_64fc** (const **Npp64fc** *pSrc, **Npp64fc** *pDst, int nLength)
64-bit complex floating point signal square root.
- **NppStatus nppsSqrt_32f_I** (**Npp32f** *pSrcDst, int nLength)
32-bit floating point signal square root.
- **NppStatus nppsSqrt_64f_I** (**Npp64f** *pSrcDst, int nLength)
64-bit floating point signal square root.
- **NppStatus nppsSqrt_32fc_I** (**Npp32fc** *pSrcDst, int nLength)
32-bit complex floating point signal square root.
- **NppStatus nppsSqrt_64fc_I** (**Npp64fc** *pSrcDst, int nLength)
64-bit complex floating point signal square root.
- **NppStatus nppsSqrt_8u_Sfs** (const **Npp8u** *pSrc, **Npp8u** *pDst, int nLength, int nScaleFactor)
8-bit unsigned char signal square root, scale, then clamp to saturated value.
- **NppStatus nppsSqrt_16u_Sfs** (const **Npp16u** *pSrc, **Npp16u** *pDst, int nLength, int nScaleFactor)
16-bit unsigned short signal square root, scale, then clamp to saturated value.
- **NppStatus nppsSqrt_16s_Sfs** (const **Npp16s** *pSrc, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal square root, scale, then clamp to saturated value.
- **NppStatus nppsSqrt_16sc_Sfs** (const **Npp16sc** *pSrc, **Npp16sc** *pDst, int nLength, int nScaleFactor)
16-bit complex signed short signal square root, scale, then clamp to saturated value.
- **NppStatus nppsSqrt_64s_Sfs** (const **Npp64s** *pSrc, **Npp64s** *pDst, int nLength, int nScaleFactor)
64-bit signed integer signal square root, scale, then clamp to saturated value.
- **NppStatus nppsSqrt_32s16s_Sfs** (const **Npp32s** *pSrc, **Npp16s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal square root, scale, then clamp to 16-bit signed integer saturated value.
- **NppStatus nppsSqrt_64s16s_Sfs** (const **Npp64s** *pSrc, **Npp16s** *pDst, int nLength, int nScaleFactor)
64-bit signed integer signal square root, scale, then clamp to 16-bit signed integer saturated value.

64-bit signed integer signal square root, scale, then clamp to 16-bit signed integer saturated value.

- **NppStatus nppsSqrt_8u_ISfs** (**Npp8u** *pSrcDst, int nLength, int nScaleFactor)
8-bit unsigned char signal square root, scale, then clamp to saturated value.
- **NppStatus nppsSqrt_16u_ISfs** (**Npp16u** *pSrcDst, int nLength, int nScaleFactor)
16-bit unsigned short signal square root, scale, then clamp to saturated value.
- **NppStatus nppsSqrt_16s_ISfs** (**Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short signal square root, scale, then clamp to saturated value.
- **NppStatus nppsSqrt_16sc_ISfs** (**Npp16sc** *pSrcDst, int nLength, int nScaleFactor)
16-bit complex signed short signal square root, scale, then clamp to saturated value.
- **NppStatus nppsSqrt_64s_ISfs** (**Npp64s** *pSrcDst, int nLength, int nScaleFactor)
64-bit signed integer signal square root, scale, then clamp to saturated value.

Cube Root Signal

Cube root of each sample of a signal.

- **NppStatus nppsCubrt_32f** (const **Npp32f** *pSrc, **Npp32f** *pDst, int nLength)
32-bit floating point signal cube root.
- **NppStatus nppsCubrt_32s16s_Sfs** (const **Npp32s** *pSrc, **Npp16s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal cube root, scale, then clamp to 16-bit signed integer saturated value.

Exponent Signal

E raised to the power of each sample of a signal.

- **NppStatus nppsExp_32f** (const **Npp32f** *pSrc, **Npp32f** *pDst, int nLength)
32-bit floating point signal exponent.
- **NppStatus nppsExp_64f** (const **Npp64f** *pSrc, **Npp64f** *pDst, int nLength)
64-bit floating point signal exponent.
- **NppStatus nppsExp_32f64f** (const **Npp32f** *pSrc, **Npp64f** *pDst, int nLength)
32-bit floating point signal exponent with 64-bit floating point result.
- **NppStatus nppsExp_32f_I** (**Npp32f** *pSrcDst, int nLength)
32-bit floating point signal exponent.
- **NppStatus nppsExp_64f_I** (**Npp64f** *pSrcDst, int nLength)
64-bit floating point signal exponent.

- **NppStatus nppsExp_16s_Sfs** (const **Npp16s** *pSrc, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal exponent, scale, then clamp to saturated value.
- **NppStatus nppsExp_32s_Sfs** (const **Npp32s** *pSrc, **Npp32s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal exponent, scale, then clamp to saturated value.
- **NppStatus nppsExp_64s_Sfs** (const **Npp64s** *pSrc, **Npp64s** *pDst, int nLength, int nScaleFactor)
64-bit signed integer signal exponent, scale, then clamp to saturated value.
- **NppStatus nppsExp_16s_ISfs** (**Npp16s** *pSrcDst, int nLength, int nScaleFactor)
16-bit signed short signal exponent, scale, then clamp to saturated value.
- **NppStatus nppsExp_32s_ISfs** (**Npp32s** *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer signal exponent, scale, then clamp to saturated value.
- **NppStatus nppsExp_64s_ISfs** (**Npp64s** *pSrcDst, int nLength, int nScaleFactor)
64-bit signed integer signal exponent, scale, then clamp to saturated value.

Natural Logarithm Signal

Natural logarithm of each sample of a signal.

- **NppStatus nppsLn_32f** (const **Npp32f** *pSrc, **Npp32f** *pDst, int nLength)
32-bit floating point signal natural logarithm.
- **NppStatus nppsLn_64f** (const **Npp64f** *pSrc, **Npp64f** *pDst, int nLength)
64-bit floating point signal natural logarithm.
- **NppStatus nppsLn_64f32f** (const **Npp64f** *pSrc, **Npp32f** *pDst, int nLength)
64-bit floating point signal natural logarithm with 32-bit floating point result.
- **NppStatus nppsLn_32f_I** (**Npp32f** *pSrcDst, int nLength)
32-bit floating point signal natural logarithm.
- **NppStatus nppsLn_64f_I** (**Npp64f** *pSrcDst, int nLength)
64-bit floating point signal natural logarithm.
- **NppStatus nppsLn_16s_Sfs** (const **Npp16s** *pSrc, **Npp16s** *pDst, int nLength, int nScaleFactor)
16-bit signed short signal natural logarithm, scale, then clamp to saturated value.
- **NppStatus nppsLn_32s_Sfs** (const **Npp32s** *pSrc, **Npp32s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal natural logarithm, scale, then clamp to saturated value.
- **NppStatus nppsLn_32s16s_Sfs** (const **Npp32s** *pSrc, **Npp16s** *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal natural logarithm, scale, then clamp to 16-bit signed short saturated value.
- **NppStatus nppsLn_16s_ISfs** (**Npp16s** *pSrcDst, int nLength, int nScaleFactor)

16-bit signed short signal natural logarithm, scale, then clamp to saturated value.

- [NppStatus nppsLn_32s_ISfs](#) ([Npp32s](#) *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer signal natural logarithm, scale, then clamp to saturated value.

Ten Times Base Ten Logarithm Signal

Ten times the decimal logarithm of each sample of a signal.

- [NppStatus npps10Log10_32s_Sfs](#) (const [Npp32s](#) *pSrc, [Npp32s](#) *pDst, int nLength, int nScaleFactor)
32-bit signed integer signal 10 times base 10 logarithm, scale, then clamp to saturated value.
- [NppStatus npps10Log10_32s_ISfs](#) ([Npp32s](#) *pSrcDst, int nLength, int nScaleFactor)
32-bit signed integer signal 10 times base 10 logarithm, scale, then clamp to saturated value.

Inverse Tangent Signal

Inverse tangent of each sample of a signal.

- [NppStatus nppsArctan_32f](#) (const [Npp32f](#) *pSrc, [Npp32f](#) *pDst, int nLength)
32-bit floating point signal inverse tangent.
- [NppStatus nppsArctan_64f](#) (const [Npp64f](#) *pSrc, [Npp64f](#) *pDst, int nLength)
64-bit floating point signal inverse tangent.
- [NppStatus nppsArctan_32f_I](#) ([Npp32f](#) *pSrcDst, int nLength)
32-bit floating point signal inverse tangent.
- [NppStatus nppsArctan_64f_I](#) ([Npp64f](#) *pSrcDst, int nLength)
64-bit floating point signal inverse tangent.

Normalize Signal

Normalize each sample of a real or complex signal using offset and division operations.

- [NppStatus nppsNormalize_32f](#) (const [Npp32f](#) *pSrc, [Npp32f](#) *pDst, int nLength, [Npp32f](#) vSub, [Npp32f](#) vDiv)
32-bit floating point signal normalize.
- [NppStatus nppsNormalize_32fc](#) (const [Npp32fc](#) *pSrc, [Npp32fc](#) *pDst, int nLength, [Npp32fc](#) vSub, [Npp32fc](#) vDiv)
32-bit complex floating point signal normalize.
- [NppStatus nppsNormalize_64f](#) (const [Npp64f](#) *pSrc, [Npp64f](#) *pDst, int nLength, [Npp64f](#) vSub, [Npp64f](#) vDiv)

64-bit floating point signal normalize.

- `NppStatus nppsNormalize_64fc` (const `Npp64fc` *pSrc, `Npp64fc` *pDst, int nLength, `Npp64fc` vSub, `Npp64f` vDiv)

64-bit complex floating point signal normalize.

- `NppStatus nppsNormalize_16s_Sfs` (const `Npp16s` *pSrc, `Npp16s` *pDst, int nLength, `Npp16s` vSub, int vDiv, int nScaleFactor)

16-bit signed short signal normalize, scale, then clamp to saturated value.

- `NppStatus nppsNormalize_16sc_Sfs` (const `Npp16sc` *pSrc, `Npp16sc` *pDst, int nLength, `Npp16sc` vSub, int vDiv, int nScaleFactor)

16-bit complex signed short signal normalize, scale, then clamp to saturated value.

Cauchy, CauchyD, and CauchyDD2 Signal

Determine Cauchy robust error function and its first and second derivatives for each sample of a signal.

- `NppStatus nppsCauchy_32f_I` (`Npp32f` *pSrcDst, int nLength, `Npp32f` nParam)

32-bit floating point signal Cauchy error calculation.

- `NppStatus nppsCauchyD_32f_I` (`Npp32f` *pSrcDst, int nLength, `Npp32f` nParam)

32-bit floating point signal Cauchy first derivative.

- `NppStatus nppsCauchyDD2_32f_I` (`Npp32f` *pSrcDst, `Npp32f` *pD2FVal, int nLength, `Npp32f` nParam)

32-bit floating point signal Cauchy first and second derivatives.

AndC

Bitwise AND of a constant and each sample of a signal.

- `NppStatus nppsAndC_8u` (const `Npp8u` *pSrc, `Npp8u` nValue, `Npp8u` *pDst, int nLength)

8-bit unsigned char signal and with constant.

- `NppStatus nppsAndC_16u` (const `Npp16u` *pSrc, `Npp16u` nValue, `Npp16u` *pDst, int nLength)

16-bit unsigned short signal and with constant.

- `NppStatus nppsAndC_32u` (const `Npp32u` *pSrc, `Npp32u` nValue, `Npp32u` *pDst, int nLength)

32-bit unsigned integer signal and with constant.

- `NppStatus nppsAndC_8u_I` (`Npp8u` nValue, `Npp8u` *pSrcDst, int nLength)

8-bit unsigned char in place signal and with constant.

- `NppStatus nppsAndC_16u_I` (`Npp16u` nValue, `Npp16u` *pSrcDst, int nLength)

16-bit unsigned short in place signal and with constant.

- `NppStatus nppsAndC_32u_I` (`Npp32u` nValue, `Npp32u` *pSrcDst, int nLength)
32-bit unsigned signed integer in place signal and with constant.

And

Sample by sample bitwise AND of samples from two signals.

- `NppStatus nppsAnd_8u` (const `Npp8u` *pSrc1, const `Npp8u` *pSrc2, `Npp8u` *pDst, int nLength)
8-bit unsigned char signal and with signal.
- `NppStatus nppsAnd_16u` (const `Npp16u` *pSrc1, const `Npp16u` *pSrc2, `Npp16u` *pDst, int nLength)
16-bit unsigned short signal and with signal.
- `NppStatus nppsAnd_32u` (const `Npp32u` *pSrc1, const `Npp32u` *pSrc2, `Npp32u` *pDst, int nLength)
32-bit unsigned integer signal and with signal.
- `NppStatus nppsAnd_8u_I` (const `Npp8u` *pSrc, `Npp8u` *pSrcDst, int nLength)
8-bit unsigned char in place signal and with signal.
- `NppStatus nppsAnd_16u_I` (const `Npp16u` *pSrc, `Npp16u` *pSrcDst, int nLength)
16-bit unsigned short in place signal and with signal.
- `NppStatus nppsAnd_32u_I` (const `Npp32u` *pSrc, `Npp32u` *pSrcDst, int nLength)
32-bit unsigned integer in place signal and with signal.

OrC

Bitwise OR of a constant and each sample of a signal.

- `NppStatus nppsOrC_8u` (const `Npp8u` *pSrc, `Npp8u` nValue, `Npp8u` *pDst, int nLength)
8-bit unsigned char signal or with constant.
- `NppStatus nppsOrC_16u` (const `Npp16u` *pSrc, `Npp16u` nValue, `Npp16u` *pDst, int nLength)
16-bit unsigned short signal or with constant.
- `NppStatus nppsOrC_32u` (const `Npp32u` *pSrc, `Npp32u` nValue, `Npp32u` *pDst, int nLength)
32-bit unsigned integer signal or with constant.
- `NppStatus nppsOrC_8u_I` (`Npp8u` nValue, `Npp8u` *pSrcDst, int nLength)
8-bit unsigned char in place signal or with constant.
- `NppStatus nppsOrC_16u_I` (`Npp16u` nValue, `Npp16u` *pSrcDst, int nLength)
16-bit unsigned short in place signal or with constant.
- `NppStatus nppsOrC_32u_I` (`Npp32u` nValue, `Npp32u` *pSrcDst, int nLength)
32-bit unsigned signed integer in place signal or with constant.

Or

Sample by sample bitwise OR of the samples from two signals.

- **NppStatus nppsOr_8u** (const **Npp8u** *pSrc1, const **Npp8u** *pSrc2, **Npp8u** *pDst, int nLength)
8-bit unsigned char signal or with signal.
- **NppStatus nppsOr_16u** (const **Npp16u** *pSrc1, const **Npp16u** *pSrc2, **Npp16u** *pDst, int nLength)
16-bit unsigned short signal or with signal.
- **NppStatus nppsOr_32u** (const **Npp32u** *pSrc1, const **Npp32u** *pSrc2, **Npp32u** *pDst, int nLength)
32-bit unsigned integer signal or with signal.
- **NppStatus nppsOr_8u_I** (const **Npp8u** *pSrc, **Npp8u** *pSrcDst, int nLength)
8-bit unsigned char in place signal or with signal.
- **NppStatus nppsOr_16u_I** (const **Npp16u** *pSrc, **Npp16u** *pSrcDst, int nLength)
16-bit unsigned short in place signal or with signal.
- **NppStatus nppsOr_32u_I** (const **Npp32u** *pSrc, **Npp32u** *pSrcDst, int nLength)
32-bit unsigned integer in place signal or with signal.

XorC

Bitwise XOR of a constant and each sample of a signal.

- **NppStatus nppsXorC_8u** (const **Npp8u** *pSrc, **Npp8u** nValue, **Npp8u** *pDst, int nLength)
8-bit unsigned char signal exclusive or with constant.
- **NppStatus nppsXorC_16u** (const **Npp16u** *pSrc, **Npp16u** nValue, **Npp16u** *pDst, int nLength)
16-bit unsigned short signal exclusive or with constant.
- **NppStatus nppsXorC_32u** (const **Npp32u** *pSrc, **Npp32u** nValue, **Npp32u** *pDst, int nLength)
32-bit unsigned integer signal exclusive or with constant.
- **NppStatus nppsXorC_8u_I** (**Npp8u** nValue, **Npp8u** *pSrcDst, int nLength)
8-bit unsigned char in place signal exclusive or with constant.
- **NppStatus nppsXorC_16u_I** (**Npp16u** nValue, **Npp16u** *pSrcDst, int nLength)
16-bit unsigned short in place signal exclusive or with constant.
- **NppStatus nppsXorC_32u_I** (**Npp32u** nValue, **Npp32u** *pSrcDst, int nLength)
32-bit unsigned signed integer in place signal exclusive or with constant.

Xor

Sample by sample bitwise XOR of the samples from two signals.

- `NppStatus nppsXor_8u` (const `Npp8u` *pSrc1, const `Npp8u` *pSrc2, `Npp8u` *pDst, int nLength)
8-bit unsigned char signal exclusive or with signal.
- `NppStatus nppsXor_16u` (const `Npp16u` *pSrc1, const `Npp16u` *pSrc2, `Npp16u` *pDst, int nLength)
16-bit unsigned short signal exclusive or with signal.
- `NppStatus nppsXor_32u` (const `Npp32u` *pSrc1, const `Npp32u` *pSrc2, `Npp32u` *pDst, int nLength)
32-bit unsigned integer signal exclusive or with signal.
- `NppStatus nppsXor_8u_I` (const `Npp8u` *pSrc, `Npp8u` *pSrcDst, int nLength)
8-bit unsigned char in place signal exclusive or with signal.
- `NppStatus nppsXor_16u_I` (const `Npp16u` *pSrc, `Npp16u` *pSrcDst, int nLength)
16-bit unsigned short in place signal exclusive or with signal.
- `NppStatus nppsXor_32u_I` (const `Npp32u` *pSrc, `Npp32u` *pSrcDst, int nLength)
32-bit unsigned integer in place signal exclusive or with signal.

Not

Bitwise NOT of each sample of a signal.

- `NppStatus nppsNot_8u` (const `Npp8u` *pSrc, `Npp8u` *pDst, int nLength)
8-bit unsigned char not signal.
- `NppStatus nppsNot_16u` (const `Npp16u` *pSrc, `Npp16u` *pDst, int nLength)
16-bit unsigned short not signal.
- `NppStatus nppsNot_32u` (const `Npp32u` *pSrc, `Npp32u` *pDst, int nLength)
32-bit unsigned integer not signal.
- `NppStatus nppsNot_8u_I` (`Npp8u` *pSrcDst, int nLength)
8-bit unsigned char in place not signal.
- `NppStatus nppsNot_16u_I` (`Npp16u` *pSrcDst, int nLength)
16-bit unsigned short in place not signal.
- `NppStatus nppsNot_32u_I` (`Npp32u` *pSrcDst, int nLength)
32-bit unsigned signed integer in place not signal.

LShiftC

Left shifts the bits of each sample of a signal by a constant amount.

- **NppStatus nppsLShiftC_8u** (const **Npp8u** *pSrc, int nValue, **Npp8u** *pDst, int nLength)
8-bit unsigned char signal left shift with constant.
- **NppStatus nppsLShiftC_16u** (const **Npp16u** *pSrc, int nValue, **Npp16u** *pDst, int nLength)
16-bit unsigned short signal left shift with constant.
- **NppStatus nppsLShiftC_16s** (const **Npp16s** *pSrc, int nValue, **Npp16s** *pDst, int nLength)
16-bit signed short signal left shift with constant.
- **NppStatus nppsLShiftC_32u** (const **Npp32u** *pSrc, int nValue, **Npp32u** *pDst, int nLength)
32-bit unsigned integer signal left shift with constant.
- **NppStatus nppsLShiftC_32s** (const **Npp32s** *pSrc, int nValue, **Npp32s** *pDst, int nLength)
32-bit signed integer signal left shift with constant.
- **NppStatus nppsLShiftC_8u_I** (int nValue, **Npp8u** *pSrcDst, int nLength)
8-bit unsigned char in place signal left shift with constant.
- **NppStatus nppsLShiftC_16u_I** (int nValue, **Npp16u** *pSrcDst, int nLength)
16-bit unsigned short in place signal left shift with constant.
- **NppStatus nppsLShiftC_16s_I** (int nValue, **Npp16s** *pSrcDst, int nLength)
16-bit signed short in place signal left shift with constant.
- **NppStatus nppsLShiftC_32u_I** (int nValue, **Npp32u** *pSrcDst, int nLength)
32-bit unsigned signed integer in place signal left shift with constant.
- **NppStatus nppsLShiftC_32s_I** (int nValue, **Npp32s** *pSrcDst, int nLength)
32-bit signed signed integer in place signal left shift with constant.

RShiftC

Right shifts the bits of each sample of a signal by a constant amount.

- **NppStatus nppsRShiftC_8u** (const **Npp8u** *pSrc, int nValue, **Npp8u** *pDst, int nLength)
8-bit unsigned char signal right shift with constant.
- **NppStatus nppsRShiftC_16u** (const **Npp16u** *pSrc, int nValue, **Npp16u** *pDst, int nLength)
16-bit unsigned short signal right shift with constant.
- **NppStatus nppsRShiftC_16s** (const **Npp16s** *pSrc, int nValue, **Npp16s** *pDst, int nLength)
16-bit signed short signal right shift with constant.
- **NppStatus nppsRShiftC_32u** (const **Npp32u** *pSrc, int nValue, **Npp32u** *pDst, int nLength)

32-bit unsigned integer signal right shift with constant.

- [NppStatus nppsRShiftC_32s](#) (const [Npp32s](#) *pSrc, int nValue, [Npp32s](#) *pDst, int nLength)
32-bit signed integer signal right shift with constant.
- [NppStatus nppsRShiftC_8u_I](#) (int nValue, [Npp8u](#) *pSrcDst, int nLength)
8-bit unsigned char in place signal right shift with constant.
- [NppStatus nppsRShiftC_16u_I](#) (int nValue, [Npp16u](#) *pSrcDst, int nLength)
16-bit unsigned short in place signal right shift with constant.
- [NppStatus nppsRShiftC_16s_I](#) (int nValue, [Npp16s](#) *pSrcDst, int nLength)
16-bit signed short in place signal right shift with constant.
- [NppStatus nppsRShiftC_32u_I](#) (int nValue, [Npp32u](#) *pSrcDst, int nLength)
32-bit unsigned signed integer in place signal right shift with constant.
- [NppStatus nppsRShiftC_32s_I](#) (int nValue, [Npp32s](#) *pSrcDst, int nLength)
32-bit signed signed integer in place signal right shift with constant.

Statistical Functions

Functions that provide global signal statistics like: average, standard deviation, minimum, etc.

- [NppStatus nppsSumGetBufferSize_32f](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 32f Sum.
- [NppStatus nppsMaxGetBufferSize_32f](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 32f Max.
- [NppStatus nppsMinGetBufferSize_32f](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 32f Min.
- [NppStatus nppsSumGetBufferSize_16s32s_Sfs](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 16s32s_Sfs.
- [NppStatus nppsMaxGetBufferSize_16s](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 16s Max.
- [NppStatus nppsMinGetBufferSize_16s](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 16s Min.
- [NppStatus nppsSumGetBufferSize_64fc](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 64fc Sum.
- [NppStatus nppsSumGetBufferSize_16s_Sfs](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 16s_Sfs Sum.
- [NppStatus nppsSumGetBufferSize_16sc_Sfs](#) (int nLength, int *hpBufferSize)

Device scratch buffer size (in bytes) for 16sc_Sfs Sum.

- [NppStatus nppsSumGetBufferSize_16sc32sc_Sfs](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 16sc32sc_Sfs Sum.
- [NppStatus nppsSumGetBufferSize_64f](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 64f Sum.
- [NppStatus nppsMaxGetBufferSize_64f](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 64f Max.
- [NppStatus nppsMinGetBufferSize_64f](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 64f Min.
- [NppStatus nppsSumGetBufferSize_32s_Sfs](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 32s_Sfs Sum.
- [NppStatus nppsMaxGetBufferSize_32s](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 32s Max.
- [NppStatus nppsMinGetBufferSize_32s](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 32s Min.
- [NppStatus nppsSumGetBufferSize_32fc](#) (int nLength, int *hpBufferSize)
Device scratch buffer size (in bytes) for 32fc Sum.
- [NppStatus nppsSum_32f](#) (const [Npp32f](#) *pSrc, int nLength, [Npp32f](#) *pSum, [NppHintAlgorithm](#) eHint, [Npp8u](#) *pDeviceBuffer)
32-bit float vector sum method
- [NppStatus nppsSum_32fc](#) (const [Npp32fc](#) *pSrc, int nLength, [Npp32fc](#) *pSum, [NppHintAlgorithm](#) eHint, [Npp8u](#) *pDeviceBuffer)
32-bit float complex vector sum method
- [NppStatus nppsSum_64f](#) (const [Npp64f](#) *pSrc, int nLength, [Npp64f](#) *pSum, [Npp8u](#) *pDeviceBuffer)
64-bit double vector sum method
- [NppStatus nppsSum_64fc](#) (const [Npp64fc](#) *pSrc, int nLength, [Npp64fc](#) *pSum, [Npp8u](#) *pDeviceBuffer)
64-bit double complex vector sum method
- [NppStatus nppsSum_16s_Sfs](#) (const [Npp16s](#) *pSrc, int nLength, [Npp16s](#) *pSum, int nScaleFactor, [Npp8u](#) *pDeviceBuffer)
16-bit short vector sum with integer scaling method
- [NppStatus nppsSum_32s_Sfs](#) (const [Npp32s](#) *pSrc, int nLength, [Npp32s](#) *pSum, int nScaleFactor, [Npp8u](#) *pDeviceBuffer)
32-bit integer vector sum with integer scaling method

- **NppStatus nppsSum_16sc_Sfs** (const **Npp16sc** *pSrc, int nLength, **Npp16sc** *pSum, int nScaleFactor, **Npp8u** *pDeviceBuffer)
16-bit short complex vector sum with integer scaling method
- **NppStatus nppsSum_16sc32sc_Sfs** (const **Npp16sc** *pSrc, int nLength, **Npp32sc** *pSum, int nScaleFactor, **Npp8u** *pDeviceBuffer)
16-bit short complex vector sum (32bit int complex) with integer scaling method
- **NppStatus nppsSum_16s32s_Sfs** (const **Npp16s** *pSrc, int nLength, **Npp32s** *pSum, int nScaleFactor, **Npp8u** *pDeviceBuffer)
16-bit integer vector sum (32bit) with integer scaling method
- **NppStatus nppsMax_16s** (const **Npp16s** *pSrc, int nLength, **Npp16s** *pMax, **Npp8u** *pDeviceBuffer)
16-bit integer vector max method
- **NppStatus nppsMax_32s** (const **Npp32s** *pSrc, int nLength, **Npp32s** *pMax, **Npp8u** *pDeviceBuffer)
32-bit integer vector max method
- **NppStatus nppsMax_32f** (const **Npp32f** *pSrc, int nLength, **Npp32f** *pMax, **Npp8u** *pDeviceBuffer)
32-bit float vector max method
- **NppStatus nppsMax_64f** (const **Npp64f** *pSrc, int nLength, **Npp64f** *pMax, **Npp8u** *pDeviceBuffer)
64-bit float vector max method
- **NppStatus nppsMin_16s** (const **Npp16s** *pSrc, int nLength, **Npp16s** *pMin, **Npp8u** *pDeviceBuffer)
16-bit integer vector min method
- **NppStatus nppsMin_32s** (const **Npp32s** *pSrc, int nLength, **Npp32s** *pMin, **Npp8u** *pDeviceBuffer)
32-bit integer vector min method
- **NppStatus nppsMin_32f** (const **Npp32f** *pSrc, int nLength, **Npp32f** *pMin, **Npp8u** *pDeviceBuffer)
32-bit integer vector min method
- **NppStatus nppsMin_64f** (const **Npp64f** *pSrc, int nLength, **Npp64f** *pMin, **Npp8u** *pDeviceBuffer)
64-bit integer vector min method
- **NppStatus nppsMinMaxGetBufferSize_8u** (int nLength, int *hpBufferSize)
Device-buffer size (in bytes) for nppsMinMax_8u.
- **NppStatus nppsMinMaxGetBufferSize_16s** (int nLength, int *hpBufferSize)
Device-buffer size (in bytes) for nppsMinMax_16s.
- **NppStatus nppsMinMaxGetBufferSize_16u** (int nLength, int *hpBufferSize)

Device-buffer size (in bytes) for nppsMinMax_16u.

- [NppStatus nppsMinMaxGetBufferSize_32s](#) (int nLength, int *hpBufferSize)
Device-buffer size (in bytes) for nppsMinMax_32s.
- [NppStatus nppsMinMaxGetBufferSize_32u](#) (int nLength, int *hpBufferSize)
Device-buffer size (in bytes) for nppsMinMax_32u.
- [NppStatus nppsMinMaxGetBufferSize_32f](#) (int nLength, int *hpBufferSize)
Device-buffer size (in bytes) for nppsMinMax_32f.
- [NppStatus nppsMinMaxGetBufferSize_64f](#) (int nLength, int *hpBufferSize)
Device-buffer size (in bytes) for nppsMinMax_64f.
- [NppStatus nppsMinMax_8u](#) (const [Npp8u](#) *pSrc, int nLength, [Npp8u](#) *pMin, [Npp8u](#) *pMax, [Npp8u](#) *pDeviceBuffer)
8-bit char vector min and max method
- [NppStatus nppsMinMax_16s](#) (const [Npp16s](#) *pSrc, int nLength, [Npp16s](#) *pMin, [Npp16s](#) *pMax, [Npp8u](#) *pDeviceBuffer)
16-bit signed short vector min and max method
- [NppStatus nppsMinMax_16u](#) (const [Npp16u](#) *pSrc, int nLength, [Npp16u](#) *pMin, [Npp16u](#) *pMax, [Npp8u](#) *pDeviceBuffer)
16-bit unsigned short vector min and max method
- [NppStatus nppsMinMax_32u](#) (const [Npp32u](#) *pSrc, int nLength, [Npp32u](#) *pMin, [Npp32u](#) *pMax, [Npp8u](#) *pDeviceBuffer)
32-bit unsigned int vector min and max method
- [NppStatus nppsMinMax_32s](#) (const [Npp32s](#) *pSrc, int nLength, [Npp32s](#) *pMin, [Npp32s](#) *pMax, [Npp8u](#) *pDeviceBuffer)
32-bit signed int vector min and max method
- [NppStatus nppsMinMax_32f](#) (const [Npp32f](#) *pSrc, int nLength, [Npp32f](#) *pMin, [Npp32f](#) *pMax, [Npp8u](#) *pDeviceBuffer)
32-bit float vector min and max method
- [NppStatus nppsMinMax_64f](#) (const [Npp64f](#) *pSrc, int nLength, [Npp64f](#) *pMin, [Npp64f](#) *pMax, [Npp8u](#) *pDeviceBuffer)
64-bit double vector min and max method

Filtering Functions

Functions that provide functionality of generating output signal based on the input signal like signal integral, etc.

- [NppStatus nppsIntegralGetBufferSize_32s](#) (int nLength, int *hpBufferSize)
- [NppStatus nppsIntegral_32s](#) (const [Npp32s](#) *pSrc, [Npp32s](#) *pDst, int nLength, [Npp8u](#) *pDeviceBuffer)

7.17.1 Function Documentation

7.17.1.1 NppStatus npps10Log10_32s_ISfs (Npp32s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal 10 times base 10 logarithm, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.2 NppStatus npps10Log10_32s_Sfs (const Npp32s * *pSrc*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal 10 times base 10 logarithm, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.3 NppStatus nppsAbs_16s (const Npp16s * *pSrc*, Npp16s * *pDst*, int *nLength*)

16-bit signed short signal absolute value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.4 NppStatus nppsAbs_16s_I (Npp16s * *pSrcDst*, int *nLength*)

16-bit signed short signal absolute value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.5 NppStatus nppsAbs_32f (const Npp32f * *pSrc*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal absolute value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.6 NppStatus nppsAbs_32f_I (Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point signal absolute value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.7 NppStatus nppsAbs_32s (const Npp32s * *pSrc*, Npp32s * *pDst*, int *nLength*)

32-bit signed integer signal absolute value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.8 NppStatus nppsAbs_32s_I (Npp32s * *pSrcDst*, int *nLength*)

32-bit signed integer signal absolute value.

Parameters:

pSrcDst In-Place Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.9 NppStatus nppsAbs_64f (const Npp64f * *pSrc*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal absolute value.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.10 NppStatus nppsAbs_64f_I (Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point signal absolute value.

Parameters:

pSrcDst In-Place Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.11 NppStatus nppsAdd_16s (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp16s * *pDst*, int *nLength*)

16-bit signed short signal add signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.
pSrc2 Source Signal Pointer. signal2 elements to be added to signal1 elements
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.12 NppStatus nppsAdd_16s32f (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp32f * *pDst*, int *nLength*)

16-bit signed short signal add signal with 32-bit floating point result, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be added to signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.13 NppStatus nppsAdd_16s32s_I (const Npp16s * *pSrc*, Npp32s * *pSrcDst*, int *nLength*)

16/32-bit signed short in place signal add signal with 32-bit signed integer results, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be added to signal1 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.14 NppStatus nppsAdd_16s_I (const Npp16s * *pSrc*, Npp16s * *pSrcDst*, int *nLength*)

16-bit signed short in place signal add signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be added to signal1 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.15 NppStatus nppsAdd_16s_ISfs (const Npp16s * *pSrc*, Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short in place signal add signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be added to signal1 elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.16 NppStatus nppsAdd_16s_Sfs (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short add signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal2 elements to be added to signal1 elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.17 NppStatus nppsAdd_16sc_ISfs (const Npp16sc * *pSrc*, Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit complex signed short in place signal add signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be added to signal1 elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.18 NppStatus nppsAdd_16sc_Sfs (const Npp16sc * *pSrc1*, const Npp16sc * *pSrc2*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed complex short add signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be added to signal1 elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.19 NppStatus nppsAdd_16u (const Npp16u * *pSrc1*, const Npp16u * *pSrc2*, Npp16u * *pDst*, int *nLength*)

16-bit unsigned short signal add signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be added to signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.20 NppStatus nppsAdd_16u_ISfs (const Npp16u * *pSrc*, Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short in place signal add signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be added to signal1 elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.21 NppStatus nppsAdd_16u_Sfs (const Npp16u * *pSrc1*, const Npp16u * *pSrc2*, Npp16u * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short add signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal2 elements to be added to signal1 elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.22 NppStatus nppsAdd_32f (const Npp32f * *pSrc1*, const Npp32f * *pSrc2*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal add signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal2 elements to be added to signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.23 NppStatus nppsAdd_32f_I (const Npp32f * *pSrc*, Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point in place signal add signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer, signal2 elements to be added to signal1 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.24 NppStatus nppsAdd_32fc (const Npp32fc * *pSrc1*, const Npp32fc * *pSrc2*, Npp32fc * *pDst*, int *nLength*)

32-bit complex floating point signal add signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be added to signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.25 NppStatus nppsAdd_32fc_I (const Npp32fc * *pSrc*, Npp32fc * *pSrcDst*, int *nLength*)

32-bit complex floating point in place signal add signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be added to signal1 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.26 NppStatus nppsAdd_32s_ISfs (const Npp32s * *pSrc*, Npp32s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer in place signal add signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be added to signal1 elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.27 NppStatus nppsAdd_32s_Sfs (const Npp32s * *pSrc1*, const Npp32s * *pSrc2*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer add signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal2 elements to be added to signal1 elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.28 NppStatus nppsAdd_32sc_ISfs (const Npp32sc * *pSrc*, Npp32sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit complex signed integer in place signal add signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be added to signal1 elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.29 NppStatus nppsAdd_32sc_Sfs (const Npp32sc * *pSrc1*, const Npp32sc * *pSrc2*, Npp32sc * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed complex integer add signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal2 elements to be added to signal1 elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.30 NppStatus nppsAdd_32u (const Npp32u * *pSrc1*, const Npp32u * *pSrc2*, Npp32u * *pDst*, int *nLength*)

32-bit unsigned int signal add signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be added to signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.31 NppStatus nppsAdd_64f (const Npp64f * *pSrc1*, const Npp64f * *pSrc2*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal add signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be added to signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.32 NppStatus nppsAdd_64f_I (const Npp64f * *pSrc*, Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point in place signal add signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be added to signal1 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.33 NppStatus nppsAdd_64fc (const Npp64fc * *pSrc1*, const Npp64fc * *pSrc2*, Npp64fc * *pDst*, int *nLength*)

64-bit complex floating point signal add signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal2 elements to be added to signal1 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.34 NppStatus nppsAdd_64fc_I (const Npp64fc * *pSrc*, Npp64fc * *pSrcDst*, int *nLength*)

64-bit complex floating point in place signal add signal, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be added to signal1 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.35 NppStatus nppsAdd_64s_Sfs (const Npp64s * *pSrc1*, const Npp64s * *pSrc2*, Npp64s * *pDst*, int *nLength*, int *nScaleFactor*)

64-bit signed integer add signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal2 elements to be added to signal1 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.36 `NppStatus nppsAdd_8u16u (const Npp8u * pSrc1, const Npp8u * pSrc2, Npp16u * pDst, int nLength)`

8-bit unsigned char signal add signal with 16-bit unsigned result, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be added to signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.37 `NppStatus nppsAdd_8u_ISfs (const Npp8u * pSrc, Npp8u * pSrcDst, int nLength, int nScaleFactor)`

8-bit unsigned char in place signal add signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be added to signal1 elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.38 `NppStatus nppsAdd_8u_Sfs (const Npp8u * pSrc1, const Npp8u * pSrc2, Npp8u * pDst, int nLength, int nScaleFactor)`

8-bit unsigned char add signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal2 elements to be added to signal1 elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.39 NppStatus nppsAddC_16s_ISfs (Npp16s *nValue*, Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short in place signal add constant, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be added to each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.40 NppStatus nppsAddC_16s_Sfs (const Npp16s * *pSrc*, Npp16s *nValue*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal add constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be added to each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.41 NppStatus nppsAddC_16sc_ISfs (Npp16sc *nValue*, Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal add constant, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be added to each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.42 NppStatus nppsAddC_16sc_Sfs (const Npp16sc * *pSrc*, Npp16sc *nValue*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal add constant, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be added to each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.43 NppStatus nppsAddC_16u_ISfs (Npp16u *nValue*, Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short in place signal add constant, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be added to each vector element

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.44 NppStatus nppsAddC_16u_Sfs (const Npp16u * *pSrc*, Npp16u *nValue*, Npp16u * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short vector add constant, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be added to each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.45 NppStatus nppsAddC_32f (const Npp32f * *pSrc*, Npp32f *nValue*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal add constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be added to each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.46 NppStatus nppsAddC_32f_I (Npp32f *nValue*, Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point in place signal add constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be added to each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.47 NppStatus nppsAddC_32fc (const Npp32fc * *pSrc*, Npp32fc *nValue*, Npp32fc * *pDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) signal add constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be added to each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.48 NppStatus nppsAddC_32fc_I (Npp32fc *nValue*, Npp32fc * *pSrcDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal add constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be added to each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.49 NppStatus nppsAddC_32s_ISfs (Npp32s *nValue*, Npp32s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer in place signal add constant and scale.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be added to each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.50 NppStatus nppsAddC_32s_Sfs (const Npp32s * *pSrc*, Npp32s *nValue*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integersignal add constant and scale.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be added to each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.51 **NppStatus nppsAddC_32sc_ISfs** (Npp32sc *nValue*, Npp32sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal add constant and scale.

Parameters:

pSrcDst [In-Place Signal Pointer](#).
nValue Constant value to be added to each vector element
nLength [Signal Length](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.52 **NppStatus nppsAddC_32sc_Sfs** (const Npp32sc * *pSrc*, Npp32sc *nValue*, Npp32sc * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) signal add constant and scale.

Parameters:

pSrc [Source Signal Pointer](#).
nValue Constant value to be added to each vector element
pDst [Destination Signal Pointer](#).
nLength [Signal Length](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.53 **NppStatus nppsAddC_64f** (const Npp64f * *pSrc*, Npp64f *nValue*, Npp64f * *pDst*, int *nLength*)

64-bit floating pointsignal add constant.

Parameters:

pSrc [Source Signal Pointer](#).
nValue Constant value to be added to each vector element
pDst [Destination Signal Pointer](#).
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.54 NppStatus nppsAddC_64f_I (Npp64f nValue, Npp64f * pSrcDst, int nLength)

64-bit floating point, in place signal add constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be added to each vector element

nLength Length of the vectors, number of items.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.55 NppStatus nppsAddC_64fc (const Npp64fc * pSrc, Npp64fc nValue, Npp64fc * pDst, int nLength)

64-bit floating point complex number (64 bit real, 64 bit imaginary) signal add constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be added to each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.56 NppStatus nppsAddC_64fc_I (Npp64fc nValue, Npp64fc * pSrcDst, int nLength)

64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal add constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be added to each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.57 NppStatus nppsAddC_8u_ISfs (Npp8u nValue, Npp8u * pSrcDst, int nLength, int nScaleFactor)

8-bit unsigned char in place signal add constant, scale, then clamp to saturated value

Parameters:

pSrcDst [In-Place Signal Pointer](#).
nValue Constant value to be added to each vector element
nLength [Signal Length](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.58 `NppStatus nppsAddC_8u_Sfs (const Npp8u *pSrc, Npp8u nValue, Npp8u *pDst, int nLength, int nScaleFactor)`

8-bit unsigned charvector add constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).
nValue Constant value to be added to each vector element
pDst [Destination Signal Pointer](#).
nLength [Signal Length](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.59 `NppStatus nppsAddProduct_16s32s_Sfs (const Npp16s *pSrc1, const Npp16s *pSrc2, Npp32s *pDst, int nLength, int nScaleFactor)`

16-bit signed short signal add product of source signal1 times source signal2 to 32-bit signed integer destination signal, with scaling, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).
pSrc2 [Source Signal Pointer](#).
pDst [Destination Signal Pointer](#). product of source1 and source2 signal elements to be added to destination elements
nLength [Signal Length](#).
nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.60 **NppStatus nppsAddProduct_16s_Sfs** (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal add product of source signal1 times source signal2 to destination signal, with scaling, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#).

pDst [Destination Signal Pointer](#). product of source1 and source2 signal elements to be added to destination elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.61 **NppStatus nppsAddProduct_32f** (const Npp32f * *pSrc1*, const Npp32f * *pSrc2*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal add product of source signal times destination signal to destination signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#).

pDst [Destination Signal Pointer](#). product of source1 and source2 signal elements to be added to destination elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.62 **NppStatus nppsAddProduct_32fc** (const Npp32fc * *pSrc1*, const Npp32fc * *pSrc2*, Npp32fc * *pDst*, int *nLength*)

32-bit complex floating point signal add product of source signal times destination signal to destination signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#).

pDst [Destination Signal Pointer](#). product of source1 and source2 signal elements to be added to destination elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.63 **NppStatus nppsAddProduct_32s_Sfs** (const Npp32s * *pSrc1*, const Npp32s * *pSrc2*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed short signal add product of source signal1 times source signal2 to destination signal, with scaling, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#).

pDst [Destination Signal Pointer](#). product of source1 and source2 signal elements to be added to destination elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.64 **NppStatus nppsAddProduct_64f** (const Npp64f * *pSrc1*, const Npp64f * *pSrc2*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal add product of source signal times destination signal to destination signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#).

pDst [Destination Signal Pointer](#). product of source1 and source2 signal elements to be added to destination elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.65 **NppStatus nppsAddProduct_64fc** (const Npp64fc * *pSrc1*, const Npp64fc * *pSrc2*, Npp64fc * *pDst*, int *nLength*)

64-bit complex floating point signal add product of source signal times destination signal to destination signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#).

pDst [Destination Signal Pointer](#). product of source1 and source2 signal elements to be added to destination elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.66 NppStatus nppsAddProductC_16s_ISfs (Npp16s *nValue*, Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short in place signal add product of signal times constant to destination signal, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.67 NppStatus nppsAddProductC_16s_Sfs (const Npp16s * *pSrc*, Npp16s *nValue*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal add product of signal times constant to destination signal, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.68 NppStatus nppsAddProductC_16sc_ISfs (Npp16sc *nValue*, Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal add product of signal times constant to destination signal, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.69 **NppStatus nppsAddProductC_16sc_Sfs** (const Npp16sc * *pSrc*, Npp16sc *nValue*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal add product of signal times constant to destination signal, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.70 **NppStatus nppsAddProductC_16u_ISfs** (Npp16u *nValue*, Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short in place signal add product of signal times constant to destination signal, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.71 **NppStatus nppsAddProductC_16u_Sfs** (const Npp16u * *pSrc*, Npp16u *nValue*, Npp16u * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short add product of signal times constant to destination signal, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.72 NppStatus nppsAddProductC_32f (const Npp32f * *pSrc*, Npp32f *nValue*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal add product of signal times constant to destination signal.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be multiplied by each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.73 NppStatus nppsAddProductC_32f_I (Npp32f *nValue*, Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point in place signal add product of signal times constant to destination signal.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be multiplied by each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.74 NppStatus nppsAddProductC_32fc (const Npp32fc * *pSrc*, Npp32fc *nValue*, Npp32fc * *pDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) signal add product of signal times constant to destination signal.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be multiplied by each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.75 **NppStatus nppsAddProductC_32fc_I** (Npp32fc *nValue*, Npp32fc * *pSrcDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal add product of signal times constant to destination signal.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.76 **NppStatus nppsAddProductC_32s_ISfs** (Npp32s *nValue*, Npp32s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer in place signal add product of signal times constant to destination signal and scale.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.77 **NppStatus nppsAddProductC_32s_Sfs** (const Npp32s * *pSrc*, Npp32s *nValue*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal add product of signal times constant to destination signal and scale.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.78 **NppStatus nppsAddProductC_32sc_ISfs** (Npp32sc *nValue*, Npp32sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal add product of signal times constant to destination signal and scale.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.79 **NppStatus nppsAddProductC_32sc_Sfs** (const Npp32sc * *pSrc*, Npp32sc *nValue*, Npp32sc * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) signal add product of signal times constant to destination signal and scale.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.80 **NppStatus nppsAddProductC_64f** (const Npp64f * *pSrc*, Npp64f *nValue*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal add product of signal times constant to destination signal.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.81 NppStatus nppsAddProductC_64f_I (Npp64f *nValue*, Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point, in place signal add product of signal times constant to destination signal.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength Length of the vectors, number of items.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.82 NppStatus nppsAddProductC_64fc (const Npp64fc * *pSrc*, Npp64fc *nValue*, Npp64fc * *pDst*, int *nLength*)

64-bit floating point complex number (64 bit real, 64 bit imaginary) signal add product of signal times constant to destination signal.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.83 NppStatus nppsAddProductC_64fc_I (Npp64fc *nValue*, Npp64fc * *pSrcDst*, int *nLength*)

64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal add product of signal times constant to destination signal.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.84 NppStatus nppsAddProductC_8u_ISfs (Npp8u *nValue*, Npp8u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char in place signal add product of signal times constant to destination signal, scale, then clamp to saturated value

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.85 NppStatus nppsAddProductC_8u_Sfs (const Npp8u * *pSrc*, Npp8u *nValue*, Npp8u * *pDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char add product of signal times constant to destination signal, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.86 NppStatus nppsAnd_16u (const Npp16u * *pSrc1*, const Npp16u * *pSrc2*, Npp16u * *pDst*, int *nLength*)

16-bit unsigned short signal and with signal.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal2 elements to be anded with signal1 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.87 NppStatus nppsAnd_16u_I (const Npp16u * *pSrc*, Npp16u * *pSrcDst*, int *nLength*)

16-bit unsigned short in place signal and with signal.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be anded with signal1 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.88 NppStatus nppsAnd_32u (const Npp32u * *pSrc1*, const Npp32u * *pSrc2*, Npp32u * *pDst*, int *nLength*)

32-bit unsigned integer signal and with signal.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal2 elements to be anded with signal1 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.89 NppStatus nppsAnd_32u_I (const Npp32u * *pSrc*, Npp32u * *pSrcDst*, int *nLength*)

32-bit unsigned integer in place signal and with signal.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be anded with signal1 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.90 NppStatus nppsAnd_8u (const Npp8u * *pSrc1*, const Npp8u * *pSrc2*, Npp8u * *pDst*, int *nLength*)

8-bit unsigned char signal and with signal.

Parameters:

pSrc1 Source Signal Pointer.
pSrc2 Source Signal Pointer. signal2 elements to be anded with signal1 elements
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.91 NppStatus nppsAnd_8u_I (const Npp8u * pSrc, Npp8u * pSrcDst, int nLength)

8-bit unsigned char in place signal and with signal.

Parameters:

pSrc Source Signal Pointer.
pSrcDst In-Place Signal Pointer. signal2 elements to be anded with signal1 elements
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.92 NppStatus nppsAndC_16u (const Npp16u * pSrc, Npp16u nValue, Npp16u * pDst, int nLength)

16-bit unsigned short signal and with constant.

Parameters:

pSrc Source Signal Pointer.
nValue Constant value to be anded with each vector element
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.93 NppStatus nppsAndC_16u_I (Npp16u nValue, Npp16u * pSrcDst, int nLength)

16-bit unsigned short in place signal and with constant.

Parameters:

pSrcDst In-Place Signal Pointer.
nValue Constant value to be anded with each vector element
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.94 NppStatus nppsAndC_32u (const Npp32u * *pSrc*, Npp32u *nValue*, Npp32u * *pDst*, int *nLength*)

32-bit unsigned integer signal and with constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be anded with each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.95 NppStatus nppsAndC_32u_I (Npp32u *nValue*, Npp32u * *pSrcDst*, int *nLength*)

32-bit unsigned signed integer in place signal and with constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be anded with each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.96 NppStatus nppsAndC_8u (const Npp8u * *pSrc*, Npp8u *nValue*, Npp8u * *pDst*, int *nLength*)

8-bit unsigned char signal and with constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be anded with each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.97 NppStatus nppsAndC_8u_I (Npp8u *nValue*, Npp8u * *pSrcDst*, int *nLength*)

8-bit unsigned char in place signal and with constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be anded with each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.98 NppStatus nppsArctan_32f (const Npp32f * *pSrc*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal inverse tangent.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.99 NppStatus nppsArctan_32f_I (Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point signal inverse tangent.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.100 NppStatus nppsArctan_64f (const Npp64f * *pSrc*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal inverse tangent.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.101 NppStatus nppsArctan_64f_I (Npp64f * pSrcDst, int nLength)

64-bit floating point signal inverse tangent.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.102 NppStatus nppsCauchy_32f_I (Npp32f * pSrcDst, int nLength, Npp32f nParam)

32-bit floating point signal Cauchy error calculation.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nParam constant used in Cauchy formula

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.103 NppStatus nppsCauchyD_32f_I (Npp32f * pSrcDst, int nLength, Npp32f nParam)

32-bit floating point signal Cauchy first derivative.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nParam constant used in Cauchy formula

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.104 NppStatus nppsCauchyDD2_32f_I (Npp32f * pSrcDst, Npp32f * pD2FVal, int nLength, Npp32f nParam)

32-bit floating point signal Cauchy first and second derivatives.

Parameters:

pSrcDst In-Place Signal Pointer.

pD2FVal Source Signal Pointer. This signal contains the second derivative of the source signal.

nLength Signal Length.

nParam constant used in Cauchy formula

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.105 NppStatus nppsCopy_16s (const Npp16s * *pSrc*, Npp16s * *pDst*, int *len*)

16-bit signed short, vector copy method.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

len Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.106 NppStatus nppsCopy_16sc (const Npp16sc * *pSrc*, Npp16sc * *pDst*, int *len*)

16-bit complex short, vector copy method.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

len Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.107 NppStatus nppsCopy_32f (const Npp32f * *pSrc*, Npp32f * *pDst*, int *len*)

32-bit float, vector copy method.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

len Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.108 NppStatus nppsCopy_32fc (const Npp32fc * *pSrc*, Npp32fc * *pDst*, int *len*)

32-bit complex float, vector copy method.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

len Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.109 NppStatus nppsCopy_32s (const Npp32s * *pSrc*, Npp32s * *pDst*, int *nLength*)

32-bit signed integer, vector copy method.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.110 NppStatus nppsCopy_32sc (const Npp32sc * *pSrc*, Npp32sc * *pDst*, int *len*)

32-bit complex signed integer, vector copy method.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

len Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.111 NppStatus nppsCopy_64fc (const Npp64fc * *pSrc*, Npp64fc * *pDst*, int *len*)

64-bit complex double, vector copy method.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

len Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.112 NppStatus nppsCopy_64s (const Npp64s * *pSrc*, Npp64s * *pDst*, int *len*)

64-bit signed integer, vector copy method.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

len Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.113 NppStatus nppsCopy_64sc (const Npp64sc * *pSrc*, Npp64sc * *pDst*, int *len*)

64-bit complex signed integer, vector copy method.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

len Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.114 NppStatus nppsCopy_8u (const Npp8u * *pSrc*, Npp8u * *pDst*, int *len*)

8-bit unsigned char, vector copy method

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

len Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.115 NppStatus nppsCubrt_32f (const Npp32f * *pSrc*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal cube root.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.116 NppStatus nppsCubrt_32s16s_Sfs (const Npp32s * *pSrc*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal cube root, scale, then clamp to 16-bit signed integer saturated value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.117 NppStatus nppsDiv_16s_ISfs (const Npp16s * *pSrc*, Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short in place signal divide signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal1 divisor elements to be divided into signal2 dividend elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.118 NppStatus nppsDiv_16s_Sfs (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal divide signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal1 divisor elements to be divided into signal2 dividend elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.119 NppStatus nppsDiv_16sc_ISfs (const Npp16sc * *pSrc*, Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit complex signed short in place signal divide signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal1 divisor elements to be divided into signal2 dividend elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.120 NppStatus nppsDiv_16sc_Sfs (const Npp16sc * *pSrc1*, const Npp16sc * *pSrc2*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed complex short signal divide signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal1 divisor elements to be divided into signal2 dividend elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.121 **NppStatus nppsDiv_16u_ISfs** (const Npp16u * *pSrc*, Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short in place signal divide signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#), signal1 divisor elements to be divided into signal2 dividend elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.122 **NppStatus nppsDiv_16u_Sfs** (const Npp16u * *pSrc1*, const Npp16u * *pSrc2*, Npp16u * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short signal divide signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal1 divisor elements to be divided into signal2 dividend elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.123 **NppStatus nppsDiv_32f** (const Npp32f * *pSrc1*, const Npp32f * *pSrc2*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal divide signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal1 divisor elements to be divided into signal2 dividend elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.124 NppStatus nppsDiv_32f_I (const Npp32f * pSrc, Npp32f * pSrcDst, int nLength)

32-bit floating point in place signal divide signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal1 divisor elements to be divided into signal2 dividend elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.125 NppStatus nppsDiv_32fc (const Npp32fc * pSrc1, const Npp32fc * pSrc2, Npp32fc * pDst, int nLength)

32-bit complex floating point signal divide signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal1 divisor elements to be divided into signal2 dividend elements.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.126 NppStatus nppsDiv_32fc_I (const Npp32fc * pSrc, Npp32fc * pSrcDst, int nLength)

32-bit complex floating point in place signal divide signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal1 divisor elements to be divided into signal2 dividend elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.127 NppStatus nppsDiv_32s16s_Sfs (const Npp16s * pSrc1, const Npp32s * pSrc2, Npp16s * pDst, int nLength, int nScaleFactor)

32-bit signed integer signal divided by 16-bit signed short signal, scale, then clamp to 16-bit signed short saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal1 divisor elements to be divided into signal2 dividend elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.128 `NppStatus nppsDiv_32s_ISfs (const Npp32s * pSrc, Npp32s * pSrcDst, int nLength, int nScaleFactor)`

32-bit signed integer in place signal divide signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal1 divisor elements to be divided into signal2 dividend elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.129 `NppStatus nppsDiv_32s_Sfs (const Npp32s * pSrc1, const Npp32s * pSrc2, Npp32s * pDst, int nLength, int nScaleFactor)`

32-bit signed integer signal divide signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal1 divisor elements to be divided into signal2 dividend elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.130 NppStatus nppsDiv_64f (const Npp64f * *pSrc1*, const Npp64f * *pSrc2*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal divide signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal1 divisor elements to be divided into signal2 dividend elements.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.131 NppStatus nppsDiv_64f_I (const Npp64f * *pSrc*, Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point in place signal divide signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal1 divisor elements to be divided into signal2 dividend elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.132 NppStatus nppsDiv_64fc (const Npp64fc * *pSrc1*, const Npp64fc * *pSrc2*, Npp64fc * *pDst*, int *nLength*)

64-bit complex floating point signal divide signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal1 divisor elements to be divided into signal2 dividend elements.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.133 NppStatus nppsDiv_64fc_I (const Npp64fc * *pSrc*, Npp64fc * *pSrcDst*, int *nLength*)

64-bit complex floating point in place signal divide signal, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 divisor elements to be divided into signal2 dividend elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.134 NppStatus nppsDiv_8u_ISfs (const Npp8u * *pSrc*, Npp8u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char in place signal divide signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 divisor elements to be divided into signal2 dividend elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.135 NppStatus nppsDiv_8u_Sfs (const Npp8u * *pSrc1*, const Npp8u * *pSrc2*, Npp8u * *pDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char signal divide signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal1 divisor elements to be divided into signal2 dividend elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.136 NppStatus nppsDiv_Round_16s_ISfs (const Npp16s * *pSrc*, Npp16s * *pSrcDst*, int *nLength*, NppRoundMode *nRndMode*, int *nScaleFactor*)

16-bit signed short in place signal divide signal, with scaling, rounding then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 divisor elements to be divided into signal2 dividend elements

nLength [Signal Length](#).

nRndMode various rounding modes.

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.137 NppStatus nppsDiv_Round_16s_Sfs (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp16s * *pDst*, int *nLength*, NppRoundMode *nRndMode*, int *nScaleFactor*)

16-bit signed short signal divide signal, scale, round, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal1 divisor elements to be divided into signal2 dividend elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nRndMode various rounding modes.

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.138 NppStatus nppsDiv_Round_16u_ISfs (const Npp16u * *pSrc*, Npp16u * *pSrcDst*, int *nLength*, NppRoundMode *nRndMode*, int *nScaleFactor*)

16-bit unsigned short in place signal divide signal, with scaling, rounding then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 divisor elements to be divided into signal2 dividend elements

nLength [Signal Length](#).

nRndMode various rounding modes.

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.139 NppStatus nppsDiv_Round_16u_Sfs (const Npp16u * *pSrc1*, const Npp16u * *pSrc2*, Npp16u * *pDst*, int *nLength*, NppRoundMode *nRndMode*, int *nScaleFactor*)

16-bit unsigned short signal divide signal, scale, round, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal1 divisor elements to be divided into signal2 dividend elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nRndMode various rounding modes.

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.140 NppStatus nppsDiv_Round_8u_ISfs (const Npp8u * *pSrc*, Npp8u * *pSrcDst*, int *nLength*, NppRoundMode *nRndMode*, int *nScaleFactor*)

8-bit unsigned char in place signal divide signal, with scaling, rounding then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#), signal1 divisor elements to be divided into signal2 dividend elements

nLength [Signal Length](#).

nRndMode various rounding modes.

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.141 NppStatus nppsDiv_Round_8u_Sfs (const Npp8u * *pSrc1*, const Npp8u * *pSrc2*, Npp8u * *pDst*, int *nLength*, NppRoundMode *nRndMode*, int *nScaleFactor*)

8-bit unsigned char signal divide signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal1 divisor elements to be divided into signal2 dividend elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nRndMode various rounding modes.

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.142 NppStatus nppsDivC_16s_ISfs (Npp16s *nValue*, Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short in place signal divided by constant, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be divided into each vector element

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.143 NppStatus nppsDivC_16s_Sfs (const Npp16s * *pSrc*, Npp16s *nValue*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be divided into each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.144 NppStatus nppsDivC_16sc_ISfs (Npp16sc *nValue*, Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal divided by constant, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be divided into each vector element

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.145 NppStatus nppsDivC_16sc_Sfs (const Npp16sc * *pSrc*, Npp16sc *nValue*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be divided into each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.146 NppStatus nppsDivC_16u_ISfs (Npp16u *nValue*, Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short in place signal divided by constant, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be divided into each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.147 NppStatus nppsDivC_16u_Sfs (const Npp16u * *pSrc*, Npp16u *nValue*, Npp16u * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short signal divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be divided into each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.148 NppStatus nppsDivC_32f (const Npp32f * *pSrc*, Npp32f *nValue*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal divided by constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be divided into each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.149 NppStatus nppsDivC_32f_I (Npp32f *nValue*, Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point in place signal divided by constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be divided into each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.150 NppStatus nppsDivC_32fc (const Npp32fc * *pSrc*, Npp32fc *nValue*, Npp32fc * *pDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) signal divided by constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be divided into each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.151 NppStatus nppsDivC_32fc_I (Npp32fc *nValue*, Npp32fc * *pSrcDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal divided by constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be divided into each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.152 NppStatus nppsDivC_32s_ISfs (Npp32s *nValue*, Npp32s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer in place signal divided by constant and scale.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be divided into each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.153 NppStatus nppsDivC_32s_Sfs (const Npp32s * *pSrc*, Npp32s *nValue*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal divided by constant and scale.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be divided into each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.154 NppStatus nppsDivC_32sc_ISfs (Npp32sc *nValue*, Npp32sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal divided by constant and scale.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be divided into each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.155 NppStatus nppsDivC_32sc_Sfs (const Npp32sc * *pSrc*, Npp32sc *nValue*, Npp32sc * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) signal divided by constant and scale.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be divided into each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.156 NppStatus nppsDivC_64f (const Npp64f * *pSrc*, Npp64f *nValue*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal divided by constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be divided into each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.157 NppStatus nppsDivC_64f_I (Npp64f *nValue*, Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point in place signal divided by constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be divided into each vector element

nLength Length of the vectors, number of items.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.158 NppStatus nppsDivC_64fc (const Npp64fc * *pSrc*, Npp64fc *nValue*, Npp64fc * *pDst*, int *nLength*)

64-bit floating point complex number (64 bit real, 64 bit imaginary) signal divided by constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be divided into each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.159 NppStatus nppsDivC_64fc_I (Npp64fc *nValue*, Npp64fc * *pSrcDst*, int *nLength*)

64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal divided by constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be divided into each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.160 NppStatus nppsDivC_8u_ISfs (Npp8u *nValue*, Npp8u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char in place signal divided by constant, scale, then clamp to saturated value

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be divided into each vector element

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.161 `NppStatus nppsDivC_8u_Sfs (const Npp8u * pSrc, Npp8u nValue, Npp8u * pDst, int nLength, int nScaleFactor)`

8-bit unsigned char signal divided by constant, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be divided into each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.162 `NppStatus nppsDivCRev_16s (const Npp16s * pSrc, Npp16s nValue, Npp16s * pDst, int nLength)`

16-bit signed short constant divided by signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be divided by each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.163 NppStatus nppsDivCRev_16s_I (Npp16s *nValue*, Npp16s * *pSrcDst*, int *nLength*)

16-bit signed short in place constant divided by signal, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be divided by each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.164 NppStatus nppsDivCRev_16u (const Npp16u * *pSrc*, Npp16u *nValue*, Npp16u * *pDst*, int *nLength*)

16-bit unsigned short signal divided by constant, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be divided by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.165 NppStatus nppsDivCRev_16u_I (Npp16u *nValue*, Npp16u * *pSrcDst*, int *nLength*)

16-bit unsigned short in place constant divided by signal, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be divided by each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.166 NppStatus nppsDivCRev_32f (const Npp32f * *pSrc*, Npp32f *nValue*, Npp32f * *pDst*, int *nLength*)

32-bit floating point constant divided by signal.

Parameters:

pSrc Source Signal Pointer.
nValue Constant value to be divided by each vector element
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.167 NppStatus nppsDivCRev_32f_I (Npp32f *nValue*, Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point in place constant divided by signal.

Parameters:

pSrcDst In-Place Signal Pointer.
nValue Constant value to be divided by each vector element
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.168 NppStatus nppsDivCRev_32s (const Npp32s * *pSrc*, Npp32s *nValue*, Npp32s * *pDst*, int *nLength*)

32-bit signed integer constant divided by signal.

Parameters:

pSrc Source Signal Pointer.
nValue Constant value to be divided by each vector element
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.169 NppStatus nppsDivCRev_32s_I (Npp32s *nValue*, Npp32s * *pSrcDst*, int *nLength*)

32-bit signed integer in place constant divided by signal.

Parameters:

pSrcDst In-Place Signal Pointer.
nValue Constant value to be divided by each vector element
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.170 NppStatus nppsDivCRev_64f (const Npp64f * *pSrc*, Npp64f *nValue*, Npp64f * *pDst*, int *nLength*)

64-bit floating point constant divided by signal.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be divided by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.171 NppStatus nppsDivCRev_64f_I (Npp64f *nValue*, Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point in place constant divided by signal.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be divided by each vector element

nLength Length of the vectors, number of items.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.172 NppStatus nppsDivCRev_8u (const Npp8u * *pSrc*, Npp8u *nValue*, Npp8u * *pDst*, int *nLength*)

8-bit unsigned char signal divided by constant, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be divided by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.173 NppStatus nppsDivCRev_8u_I (Npp8u *nValue*, Npp8u * *pSrcDst*, int *nLength*)

8-bit unsigned char signal in place constant divided by signal, scale, then clamp to saturated value

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be divided by each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.174 NppStatus nppsExp_16s_ISfs (Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal exponent, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.175 NppStatus nppsExp_16s_Sfs (const Npp16s * *pSrc*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal exponent, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.176 NppStatus nppsExp_32f (const Npp32f * *pSrc*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal exponent.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.177 NppStatus nppsExp_32f64f (const Npp32f * pSrc, Npp64f * pDst, int nLength)

32-bit floating point signal exponent with 64-bit floating point result.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.178 NppStatus nppsExp_32f_I (Npp32f * pSrcDst, int nLength)

32-bit floating point signal exponent.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.179 NppStatus nppsExp_32s_ISfs (Npp32s * pSrcDst, int nLength, int nScaleFactor)

32-bit signed integer signal exponent, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.180 NppStatus nppsExp_32s_Sfs (const Npp32s * *pSrc*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal exponent, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.181 NppStatus nppsExp_64f (const Npp64f * *pSrc*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal exponent.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.182 NppStatus nppsExp_64f_I (Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point signal exponent.

Parameters:

pSrcDst In-Place Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.183 NppStatus nppsExp_64s_ISfs (Npp64s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

64-bit signed integer signal exponent, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.184 `NppStatus nppsExp_64s_Sfs (const Npp64s * pSrc, Npp64s * pDst, int nLength, int nScaleFactor)`

64-bit signed integer signal exponent, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.185 `void nppsFree (void * pValues)`

Free method for any 2D allocated memory.

This method should be used to free memory allocated with any of the `nppiMalloc_<modifier>` methods.

Parameters:

pValues A pointer to memory allocated using `nppiMalloc_<modifier>`.

7.17.1.186 `NppStatus nppsIntegral_32s (const Npp32s * pSrc, Npp32s * pDst, int nLength, Npp8u * pDeviceBuffer)`

7.17.1.187 `NppStatus nppsIntegralGetBufferSize_32s (int nLength, int * hpBufferSize)`

7.17.1.188 `NppStatus nppsLn_16s_ISfs (Npp16s * pSrcDst, int nLength, int nScaleFactor)`

16-bit signed short signal natural logarithm, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.189 NppStatus nppsLn_16s_Sfs (const Npp16s * *pSrc*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal natural logarithm, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.190 NppStatus nppsLn_32f (const Npp32f * *pSrc*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal natural logarithm.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.191 NppStatus nppsLn_32f_I (Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point signal natural logarithm.

Parameters:

pSrcDst In-Place Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.192 NppStatus nppsLn_32s16s_Sfs (const Npp32s * *pSrc*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal natural logarithm, scale, then clamp to 16-bit signed short saturated value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.193 NppStatus nppsLn_32s_ISfs (Npp32s * pSrcDst, int nLength, int nScaleFactor)

32-bit signed integer signal natural logarithm, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.194 NppStatus nppsLn_32s_Sfs (const Npp32s * pSrc, Npp32s * pDst, int nLength, int nScaleFactor)

32-bit signed integer signal natural logarithm, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.195 NppStatus nppsLn_64f (const Npp64f * pSrc, Npp64f * pDst, int nLength)

64-bit floating point signal natural logarithm.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.196 NppStatus nppsLn_64f32f (const Npp64f * *pSrc*, Npp32f * *pDst*, int *nLength*)

64-bit floating point signal natural logarithm with 32-bit floating point result.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.197 NppStatus nppsLn_64f_I (Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point signal natural logarithm.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.198 NppStatus nppsLShiftC_16s (const Npp16s * *pSrc*, int *nValue*, Npp16s * *pDst*, int *nLength*)

16-bit signed short signal left shift with constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be used to left shift each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.199 NppStatus nppsLShiftC_16s_I (int *nValue*, Npp16s * *pSrcDst*, int *nLength*)

16-bit signed short in place signal left shift with constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be used to left shift each vector element
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.200 NppStatus nppsLShiftC_16u (const Npp16u * pSrc, int nValue, Npp16u * pDst, int nLength)

16-bit unsigned short signal left shift with constant.

Parameters:

pSrc [Source Signal Pointer](#).
nValue Constant value to be used to left shift each vector element
pDst [Destination Signal Pointer](#).
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.201 NppStatus nppsLShiftC_16u_I (int nValue, Npp16u * pSrcDst, int nLength)

16-bit unsigned short in place signal left shift with constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).
nValue Constant value to be used to left shift each vector element
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.202 NppStatus nppsLShiftC_32s (const Npp32s * pSrc, int nValue, Npp32s * pDst, int nLength)

32-bit signed integer signal left shift with constant.

Parameters:

pSrc [Source Signal Pointer](#).
nValue Constant value to be used to left shift each vector element
pDst [Destination Signal Pointer](#).
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.203 NppStatus nppsLShiftC_32s_I (int *nValue*, Npp32s * *pSrcDst*, int *nLength*)

32-bit signed integer in place signal left shift with constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be used to left shift each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.204 NppStatus nppsLShiftC_32u (const Npp32u * *pSrc*, int *nValue*, Npp32u * *pDst*, int *nLength*)

32-bit unsigned integer signal left shift with constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be used to left shift each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.205 NppStatus nppsLShiftC_32u_I (int *nValue*, Npp32u * *pSrcDst*, int *nLength*)

32-bit unsigned integer in place signal left shift with constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be used to left shift each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.206 NppStatus nppsLShiftC_8u (const Npp8u * *pSrc*, int *nValue*, Npp8u * *pDst*, int *nLength*)

8-bit unsigned char signal left shift with constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be used to left shift each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.207 NppStatus nppsLShiftC_8u_I (int *nValue*, Npp8u * *pSrcDst*, int *nLength*)

8-bit unsigned char in place signal left shift with constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be used to left shift each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.208 Npp16s* nppsMalloc_16s (int *nSize*)

16-bit signal allocator.

Parameters:

nSize Number of shorts in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.209 Npp16sc* nppsMalloc_16sc (int *nSize*)

16-bit complex-value signal allocator.

Parameters:

nSize Number of 16-bit complex numbers in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.210 Npp16u* nppsMalloc_16u (int nSize)

16-bit unsigned signal allocator.

Parameters:

nSize Number of unsigned shorts in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.211 Npp32f* nppsMalloc_32f (int nSize)

32-bit float signal allocator.

Parameters:

nSize Number of floats in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.212 Npp32fc* nppsMalloc_32fc (int nSize)

32-bit complex float signal allocator.

Parameters:

nSize Number of complex float values in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.213 Npp32s* nppsMalloc_32s (int nSize)

32-bit integer signal allocator.

Parameters:

nSize Number of ints in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.214 Npp32sc* nppsMalloc_32sc (int *nSize*)

32-bit complex integer signal allocator.

Parameters:

nSize Number of complex integner values in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.215 Npp32u* nppsMalloc_32u (int *nSize*)

32-bit unsigned signal allocator.

Parameters:

nSize Number of unsigned ints in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.216 Npp64f* nppsMalloc_64f (int *nSize*)

64-bit float (double) signal allocator.

Parameters:

nSize Number of doubles in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.217 Npp64fc* nppsMalloc_64fc (int *nSize*)

64-bit complex complex signal allocator.

Parameters:

nSize Number of complex double valuess in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.218 Npp64s* nppsMalloc_64s (int nSize)

64-bit long integer signal allocator.

Parameters:

nSize Number of long ints in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.219 Npp64sc* nppsMalloc_64sc (int nSize)

64-bit complex long integer signal allocator.

Parameters:

nSize Number of complex long int values in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.220 Npp8u* nppsMalloc_8u (int nSize)

8-bit unsigned signal allocator.

Parameters:

nSize Number of unsigned chars in the new signal.

Returns:

A pointer to the new signal. 0 (NULL-pointer) indicates that an error occurred during allocation.

7.17.1.221 NppStatus nppsMax_16s (const Npp16s *pSrc, int nLength, Npp16s *pMax, Npp8u *pDeviceBuffer)

16-bit integer vector max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMax Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.222 NppStatus nppsMax_32f (const Npp32f * *pSrc*, int *nLength*, Npp32f * *pMax*, Npp8u * *pDeviceBuffer*)

32-bit float vector max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMax Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.223 NppStatus nppsMax_32s (const Npp32s * *pSrc*, int *nLength*, Npp32s * *pMax*, Npp8u * *pDeviceBuffer*)

32-bit integer vector max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMax Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.224 NppStatus nppsMax_64f (const Npp64f * *pSrc*, int *nLength*, Npp64f * *pMax*, Npp8u * *pDeviceBuffer*)

64-bit float vector max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMax Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.225 NppStatus nppsMaxGetBufferSize_16s (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 16s Max.

This primitive provides the correct buffer size for nppsMax_16s.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.226 NppStatus nppsMaxGetBufferSize_32f (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 32f Max.

This primitive provides the correct buffer size for nppsMax_32f.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.227 NppStatus nppsMaxGetBufferSize_32s (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 32s Max.

This primitive provides the correct buffer size for nppsMax_32s.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.228 NppStatus nppsMaxGetBufferSize_64f (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 64f Max.

This primitive provides the correct buffer size for nppsMax_64f.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*: [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.229 **NppStatus nppsMin_16s** (const Npp16s * *pSrc*, int *nLength*, Npp16s * *pMin*, Npp8u * *pDeviceBuffer*)

16-bit integer vector min method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.230 **NppStatus nppsMin_32f** (const Npp32f * *pSrc*, int *nLength*, Npp32f * *pMin*, Npp8u * *pDeviceBuffer*)

32-bit integer vector min method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.231 **NppStatus nppsMin_32s** (const Npp32s * *pSrc*, int *nLength*, Npp32s * *pMin*, Npp8u * *pDeviceBuffer*)

32-bit integer vector min method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.232 NppStatus nppsMin_64f (const Npp64f * pSrc, int nLength, Npp64f * pMin, Npp8u * pDeviceBuffer)

64-bit integer vector min method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.233 NppStatus nppsMinGetBufferSize_16s (int nLength, int * hpBufferSize)

Device scratch buffer size (in bytes) for 16s Min.

This primitive provides the correct buffer size for nppsMin_16s.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: hpBufferSize is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.234 NppStatus nppsMinGetBufferSize_32f (int nLength, int * hpBufferSize)

Device scratch buffer size (in bytes) for 32f Min.

This primitive provides the correct buffer size for nppsMin_32f.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: hpBufferSize is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.235 NppStatus nppsMinGetBufferSize_32s (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 32s Min.

This primitive provides the correct buffer size for nppsMin_32s.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.236 NppStatus nppsMinGetBufferSize_64f (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 64f Min.

This primitive provides the correct buffer size for nppsMin_64f.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.237 NppStatus nppsMinMax_16s (const Npp16s * *pSrc*, int *nLength*, Npp16s * *pMin*, Npp16s * *pMax*, Npp8u * *pDeviceBuffer*)

16-bit signed short vector min and max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the min output result.

pMax Pointer to the max output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.238 NppStatus nppsMinMax_16u (const Npp16u * *pSrc*, int *nLength*, Npp16u * *pMin*, Npp16u * *pMax*, Npp8u * *pDeviceBuffer*)

16-bit unsigned short vector min and max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the min output result.

pMax Pointer to the max output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.239 NppStatus nppsMinMax_32f (const Npp32f * *pSrc*, int *nLength*, Npp32f * *pMin*, Npp32f * *pMax*, Npp8u * *pDeviceBuffer*)

32-bit float vector min and max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the min output result.

pMax Pointer to the max output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.240 NppStatus nppsMinMax_32s (const Npp32s * *pSrc*, int *nLength*, Npp32s * *pMin*, Npp32s * *pMax*, Npp8u * *pDeviceBuffer*)

32-bit signed int vector min and max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the min output result.

pMax Pointer to the max output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.241 `NppStatus nppsMinMax_32u (const Npp32u * pSrc, int nLength, Npp32u * pMin, Npp32u * pMax, Npp8u * pDeviceBuffer)`

32-bit unsigned int vector min and max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the min output result.

pMax Pointer to the max output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.242 `NppStatus nppsMinMax_64f (const Npp64f * pSrc, int nLength, Npp64f * pMin, Npp64f * pMax, Npp8u * pDeviceBuffer)`

64-bit double vector min and max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the min output result.

pMax Pointer to the max output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.243 `NppStatus nppsMinMax_8u (const Npp8u * pSrc, int nLength, Npp8u * pMin, Npp8u * pMax, Npp8u * pDeviceBuffer)`

8-bit char vector min and max method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pMin Pointer to the min output result.

pMax Pointer to the max output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.244 NppStatus nppsMinMaxGetBufferSize_16s (int *nLength*, int * *hpBufferSize*)

Device-buffer size (in bytes) for nppsMinMax_16s.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*.

Returns:

NPP_SUCCESS

7.17.1.245 NppStatus nppsMinMaxGetBufferSize_16u (int *nLength*, int * *hpBufferSize*)

Device-buffer size (in bytes) for nppsMinMax_16u.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*.

Returns:

NPP_SUCCESS

7.17.1.246 NppStatus nppsMinMaxGetBufferSize_32f (int *nLength*, int * *hpBufferSize*)

Device-buffer size (in bytes) for nppsMinMax_32f.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*.

Returns:

NPP_SUCCESS

7.17.1.247 NppStatus nppsMinMaxGetBufferSize_32s (int *nLength*, int * *hpBufferSize*)

Device-buffer size (in bytes) for nppsMinMax_32s.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*.

Returns:

NPP_SUCCESS

7.17.1.248 NppStatus nppsMinMaxGetBufferSize_32u (int *nLength*, int * *hpBufferSize*)

Device-buffer size (in bytes) for nppsMinMax_32u.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*.

Returns:

NPP_SUCCESS

7.17.1.249 NppStatus nppsMinMaxGetBufferSize_64f (int *nLength*, int * *hpBufferSize*)

Device-buffer size (in bytes) for nppsMinMax_64f.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*.

Returns:

NPP_SUCCESS

7.17.1.250 NppStatus nppsMinMaxGetBufferSize_8u (int *nLength*, int * *hpBufferSize*)

Device-buffer size (in bytes) for nppsMinMax_8u.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*.

Returns:

NPP_SUCCESS

7.17.1.251 NppStatus nppsMul_16s (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp16s * *pDst*, int *nLength*)

16-bit signed short signal times signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal2 elements to be multiplied by signal1 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.252 NppStatus nppsMul_16s32f (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp32f * *pDst*, int *nLength*)

16-bit signed short signal times signal with 32-bit floating point result, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be multiplied by signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.253 NppStatus nppsMul_16s32s_Sfs (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal times signal, scale, then clamp to 32-bit signed saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal2 elements to be multiplied by signal1 elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.254 NppStatus nppsMul_16s_I (const Npp16s * *pSrc*, Npp16s * *pSrcDst*, int *nLength*)

16-bit signed short in place signal times signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be multiplied by signal1 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.255 **NppStatus nppsMul_16s_ISfs** (const Npp16s * *pSrc*, Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short in place signal times signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be multiplied by signal1 elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.256 **NppStatus nppsMul_16s_Sfs** (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal times signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal2 elements to be multiplied by signal1 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.257 **NppStatus nppsMul_16sc_ISfs** (const Npp16sc * *pSrc*, Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit complex signed short in place signal times signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be multiplied by signal1 elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.258 NppStatus nppsMul_16sc_Sfs (const Npp16sc * *pSrc1*, const Npp16sc * *pSrc2*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed complex short signal times signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal2 elements to be multiplied by signal1 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.259 NppStatus nppsMul_16u16s_Sfs (const Npp16u * *pSrc1*, const Npp16s * *pSrc2*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short signal times 16-bit signed short signal, scale, then clamp to 16-bit signed saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal2 elements to be multiplied by signal1 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.260 NppStatus nppsMul_16u_ISfs (const Npp16u * *pSrc*, Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short in place signal times signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be multiplied by signal1 elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.261 `NppStatus nppsMul_16u_Sfs (const Npp16u * pSrc1, const Npp16u * pSrc2, Npp16u * pDst, int nLength, int nScaleFactor)`

16-bit unsigned short signal time signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal2 elements to be multiplied by signal1 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.262 `NppStatus nppsMul_32f (const Npp32f * pSrc1, const Npp32f * pSrc2, Npp32f * pDst, int nLength)`

32-bit floating point signal times signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal2 elements to be multiplied by signal1 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.263 `NppStatus nppsMul_32f32fc (const Npp32f * pSrc1, const Npp32fc * pSrc2, Npp32fc * pDst, int nLength)`

32-bit floating point signal times 32-bit complex floating point signal with complex 32-bit floating point result, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal2 elements to be multiplied by signal1 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.264 NppStatus nppsMul_32f32fc_I (const Npp32f * *pSrc*, Npp32fc * *pSrcDst*, int *nLength*)

32-bit complex floating point in place signal times 32-bit floating point signal, then clamp to 32-bit complex floating point saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be multiplied by signal1 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.265 NppStatus nppsMul_32f_I (const Npp32f * *pSrc*, Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point in place signal times signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be multiplied by signal1 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.266 NppStatus nppsMul_32fc (const Npp32fc * *pSrc1*, const Npp32fc * *pSrc2*, Npp32fc * *pDst*, int *nLength*)

32-bit complex floating point signal times signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be multiplied by signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.267 NppStatus nppsMul_32fc_I (const Npp32fc * *pSrc*, Npp32fc * *pSrcDst*, int *nLength*)

32-bit complex floating point in place signal times signal, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be multiplied by signal1 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.268 NppStatus nppsMul_32s32sc_ISfs (const Npp32s * *pSrc*, Npp32sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit complex signed integer in place signal times 32-bit signed integer signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be multiplied by signal1 elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.269 NppStatus nppsMul_32s32sc_Sfs (const Npp32s * *pSrc1*, const Npp32sc * *pSrc2*, Npp32sc * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal times 32-bit complex signed integer signal, scale, then clamp to 32-bit complex integer saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal2 elements to be multiplied by signal1 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.270 NppStatus nppsMul_32s_ISfs (const Npp32s * *pSrc*, Npp32s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer in place signal times signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be multiplied by signal1 elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.271 NppStatus nppsMul_32s_Sfs (const Npp32s * *pSrc1*, const Npp32s * *pSrc2*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal times signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal2 elements to be multiplied by signal1 elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.272 NppStatus nppsMul_32sc_ISfs (const Npp32sc * *pSrc*, Npp32sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit complex signed integer in place signal times signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be multiplied by signal1 elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.273 NppStatus nppsMul_32sc_Sfs (const Npp32sc * *pSrc1*, const Npp32sc * *pSrc2*, Npp32sc * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed complex integer signal times signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal2 elements to be multiplied by signal1 elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.274 NppStatus nppsMul_64f (const Npp64f * *pSrc1*, const Npp64f * *pSrc2*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal times signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal2 elements to be multiplied by signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.275 NppStatus nppsMul_64f_I (const Npp64f * *pSrc*, Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point in place signal times signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer, signal2 elements to be multiplied by signal1 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.276 NppStatus nppsMul_64fc (const Npp64fc * *pSrc1*, const Npp64fc * *pSrc2*, Npp64fc * *pDst*, int *nLength*)

64-bit complex floating point signal times signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be multiplied by signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.277 NppStatus nppsMul_64fc_I (const Npp64fc * *pSrc*, Npp64fc * *pSrcDst*, int *nLength*)

64-bit complex floating point in place signal times signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal2 elements to be multiplied by signal1 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.278 NppStatus nppsMul_8u16u (const Npp8u * *pSrc1*, const Npp8u * *pSrc2*, Npp16u * *pDst*, int *nLength*)

8-bit unsigned char signal times signal with 16-bit unsigned result, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal2 elements to be multiplied by signal1 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.279 NppStatus nppsMul_8u_ISfs (const Npp8u * *pSrc*, Npp8u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char in place signal times signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#), signal2 elements to be multiplied by signal1 elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.280 NppStatus nppsMul_8u_Sfs (const Npp8u * *pSrc1*, const Npp8u * *pSrc2*, Npp8u * *pDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char signal times signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal2 elements to be multiplied by signal1 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.281 NppStatus nppsMul_Low_32s_Sfs (const Npp32s * *pSrc1*, const Npp32s * *pSrc2*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal times signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal2 elements to be multiplied by signal1 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.282 NppStatus nppsMulC_16s_ISfs (Npp16s *nValue*, Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short in place signal times constant, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be multiplied by each vector element

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.283 NppStatus nppsMulC_16s_Sfs (const Npp16s * *pSrc*, Npp16s *nValue*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal times constant, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be multiplied by each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.284 NppStatus nppsMulC_16sc_ISfs (Npp16sc *nValue*, Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal times constant, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be multiplied by each vector element

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.285 NppStatus nppsMulC_16sc_Sfs (const Npp16sc * *pSrc*, Npp16sc *nValue*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal times constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.286 NppStatus nppsMulC_16u_ISfs (Npp16u *nValue*, Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short in place signal times constant, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.287 NppStatus nppsMulC_16u_Sfs (const Npp16u * *pSrc*, Npp16u *nValue*, Npp16u * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short signal times constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.288 NppStatus nppsMulC_32f (const Npp32f * *pSrc*, Npp32f *nValue*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal times constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be multiplied by each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.289 NppStatus nppsMulC_32f16s_Sfs (const Npp32f * *pSrc*, Npp32f *nValue*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit floating point signal times constant with output converted to 16-bit signed integer with scaling and saturation of output result.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be multiplied by each vector element

pDst Destination Signal Pointer.

nScaleFactor Integer Result Scaling.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.290 NppStatus nppsMulC_32f_I (Npp32f *nValue*, Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point in place signal times constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be multiplied by each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.291 **NppStatus nppsMulC_32fc** (const Npp32fc * *pSrc*, Npp32fc *nValue*, Npp32fc * *pDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) signal times constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.292 **NppStatus nppsMulC_32fc_I** (Npp32fc *nValue*, Npp32fc * *pSrcDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal times constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.293 **NppStatus nppsMulC_32s_ISfs** (Npp32s *nValue*, Npp32s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer in place signal times constant and scale.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.294 NppStatus nppsMulC_32s_Sfs (const Npp32s * *pSrc*, Npp32s *nValue*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal times constant and scale.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.295 NppStatus nppsMulC_32sc_ISfs (Npp32sc *nValue*, Npp32sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal times constant and scale.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.296 NppStatus nppsMulC_32sc_Sfs (const Npp32sc * *pSrc*, Npp32sc *nValue*, Npp32sc * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) signal times constant and scale.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.297 NppStatus nppsMulC_64f (const Npp64f * *pSrc*, Npp64f *nValue*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal times constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.298 NppStatus nppsMulC_64f64s_ISfs (Npp64f *nValue*, Npp64s * *pDst*, int *nLength*, int *nScaleFactor*)

64-bit floating point signal times constant with in place conversion to 64-bit signed integer and with scaling and saturation of output result.

Parameters:

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.299 NppStatus nppsMulC_64f_I (Npp64f *nValue*, Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point, in place signal times constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

nLength Length of the vectors, number of items.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.300 NppStatus nppsMulC_64fc (const Npp64fc * *pSrc*, Npp64fc *nValue*, Npp64fc * *pDst*, int *nLength*)

64-bit floating point complex number (64 bit real, 64 bit imaginary) signal times constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be multiplied by each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.301 NppStatus nppsMulC_64fc_I (Npp64fc *nValue*, Npp64fc * *pSrcDst*, int *nLength*)

64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal times constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be multiplied by each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.302 NppStatus nppsMulC_8u_ISfs (Npp8u *nValue*, Npp8u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char in place signal times constant, scale, then clamp to saturated value

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be multiplied by each vector element

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.303 **NppStatus nppsMulC_8u_Sfs** (const Npp8u * *pSrc*, Npp8u *nValue*, Npp8u * *pDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char signal times constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.304 **NppStatus nppsMulC_Low_32f16s** (const Npp32f * *pSrc*, Npp32f *nValue*, Npp16s * *pDst*, int *nLength*)

32-bit floating point signal times constant with output converted to 16-bit signed integer.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be multiplied by each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.305 **NppStatus nppsNormalize_16s_Sfs** (const Npp16s * *pSrc*, Npp16s * *pDst*, int *nLength*, Npp16s *vSub*, int *vDiv*, int *nScaleFactor*)

16-bit signed short signal normalize, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

vSub value subtracted from each signal element before division

vDiv divisor of post-subtracted signal element dividend

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.306 NppStatus nppsNormalize_16sc_Sfs (const Npp16sc * pSrc, Npp16sc * pDst, int nLength, Npp16sc vSub, int vDiv, int nScaleFactor)

16-bit complex signed short signal normalize, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

vSub value subtracted from each signal element before division

vDiv divisor of post-subtracted signal element dividend

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.307 NppStatus nppsNormalize_32f (const Npp32f * pSrc, Npp32f * pDst, int nLength, Npp32f vSub, Npp32f vDiv)

32-bit floating point signal normalize.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

vSub value subtracted from each signal element before division

vDiv divisor of post-subtracted signal element dividend

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.308 NppStatus nppsNormalize_32fc (const Npp32fc * pSrc, Npp32fc * pDst, int nLength, Npp32fc vSub, Npp32fc vDiv)

32-bit complex floating point signal normalize.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

vSub value subtracted from each signal element before division

vDiv divisor of post-subtracted signal element dividend

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.309 NppStatus nppsNormalize_64f (const Npp64f * *pSrc*, Npp64f * *pDst*, int *nLength*, Npp64f *vSub*, Npp64f *vDiv*)

64-bit floating point signal normalize.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

vSub value subtracted from each signal element before division

vDiv divisor of post-subtracted signal element dividend

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.310 NppStatus nppsNormalize_64fc (const Npp64fc * *pSrc*, Npp64fc * *pDst*, int *nLength*, Npp64fc *vSub*, Npp64fc *vDiv*)

64-bit complex floating point signal normalize.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

vSub value subtracted from each signal element before division

vDiv divisor of post-subtracted signal element dividend

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.311 NppStatus nppsNot_16u (const Npp16u * *pSrc*, Npp16u * *pDst*, int *nLength*)

16-bit unsigned short not signal.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.312 NppStatus nppsNot_16u_I (Npp16u * *pSrcDst*, int *nLength*)

16-bit unsigned short in place not signal.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.313 NppStatus nppsNot_32u (const Npp32u * *pSrc*, Npp32u * *pDst*, int *nLength*)

32-bit unsigned integer not signal.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.314 NppStatus nppsNot_32u_I (Npp32u * *pSrcDst*, int *nLength*)

32-bit unsigned signed integer in place not signal.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.315 NppStatus nppsNot_8u (const Npp8u * *pSrc*, Npp8u * *pDst*, int *nLength*)

8-bit unsigned char not signal.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.316 NppStatus nppsNot_8u_I (Npp8u * *pSrcDst*, int *nLength*)

8-bit unsigned char in place not signal.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.317 NppStatus nppsOr_16u (const Npp16u * *pSrc1*, const Npp16u * *pSrc2*, Npp16u * *pDst*, int *nLength*)

16-bit unsigned short signal or with signal.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal2 elements to be ored with signal1 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.318 NppStatus nppsOr_16u_I (const Npp16u * *pSrc*, Npp16u * *pSrcDst*, int *nLength*)

16-bit unsigned short in place signal or with signal.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be ored with signal1 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.319 NppStatus nppsOr_32u (const Npp32u * *pSrc1*, const Npp32u * *pSrc2*, Npp32u * *pDst*, int *nLength*)

32-bit unsigned integer signal or with signal.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal2 elements to be ored with signal1 elements
pDst [Destination Signal Pointer](#).
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.320 NppStatus nppsOr_32u_I (const Npp32u * pSrc, Npp32u * pSrcDst, int nLength)

32-bit unsigned integer in place signal or with signal.

Parameters:

pSrc [Source Signal Pointer](#).
pSrcDst [In-Place Signal Pointer](#). signal2 elements to be ored with signal1 elements
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.321 NppStatus nppsOr_8u (const Npp8u * pSrc1, const Npp8u * pSrc2, Npp8u * pDst, int nLength)

8-bit unsigned char signal or with signal.

Parameters:

pSrc1 [Source Signal Pointer](#).
pSrc2 [Source Signal Pointer](#). signal2 elements to be ored with signal1 elements
pDst [Destination Signal Pointer](#).
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.322 NppStatus nppsOr_8u_I (const Npp8u * pSrc, Npp8u * pSrcDst, int nLength)

8-bit unsigned char in place signal or with signal.

Parameters:

pSrc [Source Signal Pointer](#).
pSrcDst [In-Place Signal Pointer](#). signal2 elements to be ored with signal1 elements
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.323 NppStatus nppsOrC_16u (const Npp16u * *pSrc*, Npp16u *nValue*, Npp16u * *pDst*, int *nLength*)

16-bit unsigned short signal or with constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be ored with each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.324 NppStatus nppsOrC_16u_I (Npp16u *nValue*, Npp16u * *pSrcDst*, int *nLength*)

16-bit unsigned short in place signal or with constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be ored with each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.325 NppStatus nppsOrC_32u (const Npp32u * *pSrc*, Npp32u *nValue*, Npp32u * *pDst*, int *nLength*)

32-bit unsigned integer signal or with constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be ored with each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.326 NppStatus nppsOrC_32u_I (Npp32u *nValue*, Npp32u * *pSrcDst*, int *nLength*)

32-bit unsigned signed integer in place signal or with constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be ored with each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.327 NppStatus nppsOrC_8u (const Npp8u * *pSrc*, Npp8u *nValue*, Npp8u * *pDst*, int *nLength*)

8-bit unsigned char signal or with constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be ored with each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.328 NppStatus nppsOrC_8u_I (Npp8u *nValue*, Npp8u * *pSrcDst*, int *nLength*)

8-bit unsigned char in place signal or with constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be ored with each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.329 NppStatus nppsRShiftC_16s (const Npp16s * *pSrc*, int *nValue*, Npp16s * *pDst*, int *nLength*)

16-bit signed short signal right shift with constant.

Parameters:

pSrc Source Signal Pointer.
nValue Constant value to be used to right shift each vector element
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.330 NppStatus nppsRShiftC_16s_I (int *nValue*, Npp16s * *pSrcDst*, int *nLength*)

16-bit signed short in place signal right shift with constant.

Parameters:

pSrcDst In-Place Signal Pointer.
nValue Constant value to be used to right shift each vector element
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.331 NppStatus nppsRShiftC_16u (const Npp16u * *pSrc*, int *nValue*, Npp16u * *pDst*, int *nLength*)

16-bit unsigned short signal right shift with constant.

Parameters:

pSrc Source Signal Pointer.
nValue Constant value to be used to right shift each vector element
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.332 NppStatus nppsRShiftC_16u_I (int *nValue*, Npp16u * *pSrcDst*, int *nLength*)

16-bit unsigned short in place signal right shift with constant.

Parameters:

pSrcDst In-Place Signal Pointer.
nValue Constant value to be used to right shift each vector element
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.333 NppStatus nppsRShiftC_32s (const Npp32s * *pSrc*, int *nValue*, Npp32s * *pDst*, int *nLength*)

32-bit signed integer signal right shift with constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be used to right shift each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.334 NppStatus nppsRShiftC_32s_I (int *nValue*, Npp32s * *pSrcDst*, int *nLength*)

32-bit signed integer in place signal right shift with constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be used to right shift each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.335 NppStatus nppsRShiftC_32u (const Npp32u * *pSrc*, int *nValue*, Npp32u * *pDst*, int *nLength*)

32-bit unsigned integer signal right shift with constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be used to right shift each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.336 NppStatus nppsRShiftC_32u_I (int *nValue*, Npp32u * *pSrcDst*, int *nLength*)

32-bit unsigned signed integer in place signal right shift with constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).
nValue Constant value to be used to right shift each vector element
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.337 NppStatus nppsRShiftC_8u (const Npp8u * *pSrc*, int *nValue*, Npp8u * *pDst*, int *nLength*)

8-bit unsigned char signal right shift with constant.

Parameters:

pSrc [Source Signal Pointer](#).
nValue Constant value to be used to right shift each vector element
pDst [Destination Signal Pointer](#).
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.338 NppStatus nppsRShiftC_8u_I (int *nValue*, Npp8u * *pSrcDst*, int *nLength*)

8-bit unsigned char in place signal right shift with constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).
nValue Constant value to be used to right shift each vector element
nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.339 NppStatus nppsSet_16s (Npp16s *nValue*, Npp16s * *pDst*, int *nLength*)

16-bit integer, vector set method.

Parameters:

nValue Value used to initialize the vector *pDst*.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.340 NppStatus nppsSet_16sc (Npp16sc *nValue*, Npp16sc * *pDst*, int *nLength*)

16-bit integer complex, vector set method.

Parameters:

nValue Value used to initialize the vector pDst.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.341 NppStatus nppsSet_32f (Npp32f *nValue*, Npp32f * *pDst*, int *nLength*)

32-bit float, vector set method.

Parameters:

nValue Value used to initialize the vector pDst.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.342 NppStatus nppsSet_32fc (Npp32fc *nValue*, Npp32fc * *pDst*, int *nLength*)

32-bit float complex, vector set method.

Parameters:

nValue Value used to initialize the vector pDst.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.343 NppStatus nppsSet_32s (Npp32s *nValue*, Npp32s * *pDst*, int *nLength*)

32-bit integer, vector set method.

Parameters:

nValue Value used to initialize the vector pDst.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.344 NppStatus nppsSet_32sc (Npp32sc *nValue*, Npp32sc * *pDst*, int *nLength*)

32-bit integer complex, vector set method.

Parameters:

nValue Value used to initialize the vector pDst.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.345 NppStatus nppsSet_64f (Npp64f *nValue*, Npp64f * *pDst*, int *nLength*)

64-bit double, vector set method.

Parameters:

nValue Value used to initialize the vector pDst.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.346 NppStatus nppsSet_64fc (Npp64fc *nValue*, Npp64fc * *pDst*, int *nLength*)

64-bit double complex, vector set method.

Parameters:

nValue Value used to initialize the vector pDst.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.347 NppStatus nppsSet_64s (Npp64s *nValue*, Npp64s * *pDst*, int *nLength*)

64-bit long long integer, vector set method.

Parameters:

nValue Value used to initialize the vector pDst.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.348 NppStatus nppsSet_64sc (Npp64sc *nValue*, Npp64sc * *pDst*, int *nLength*)

64-bit long long integer complex, vector set method.

Parameters:

nValue Value used to initialize the vector pDst.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.349 NppStatus nppsSet_8u (Npp8u *nValue*, Npp8u * *pDst*, int *nLength*)

8-bit unsigned char, vector set method.

Parameters:

nValue Value used to initialize the vector pDst.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.350 NppStatus nppsSqr_16s_ISfs (Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal squared, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.351 NppStatus nppsSqr_16s_Sfs (const Npp16s * *pSrc*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal squared, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.352 NppStatus nppsSqr_16sc_ISfs (Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit complex signed short signal squared, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.353 NppStatus nppsSqr_16sc_Sfs (const Npp16sc * *pSrc*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit complex signed short signal squared, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.354 NppStatus nppsSqr_16u_ISfs (Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short signal squared, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.355 NppStatus nppsSqr_16u_Sfs (const Npp16u * *pSrc*, Npp16u * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short signal squared, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.356 NppStatus nppsSqr_32f (const Npp32f * *pSrc*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal squared.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.357 NppStatus nppsSqr_32f_I (Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point signal squared.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.358 NppStatus nppsSqr_32fc (const Npp32fc * *pSrc*, Npp32fc * *pDst*, int *nLength*)

32-bit complex floating point signal squared.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.359 NppStatus nppsSqr_32fc_I (Npp32fc * *pSrcDst*, int *nLength*)

32-bit complex floating point signal squared.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.360 NppStatus nppsSqr_64f (const Npp64f * *pSrc*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal squared.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.361 NppStatus nppsSqr_64f_I (Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point signal squared.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.362 NppStatus nppsSqr_64fc (const Npp64fc * *pSrc*, Npp64fc * *pDst*, int *nLength*)

64-bit complex floating point signal squared.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.363 NppStatus nppsSqr_64fc_I (Npp64fc * *pSrcDst*, int *nLength*)

64-bit complex floating point signal squared.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.364 NppStatus nppsSqr_8u_ISfs (Npp8u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char signal squared, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.365 NppStatus nppsSqr_8u_Sfs (const Npp8u * *pSrc*, Npp8u * *pDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char signal squared, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.366 NppStatus nppsSqrt_16s_ISfs (Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal square root, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.367 NppStatus nppsSqrt_16s_Sfs (const Npp16s * *pSrc*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal square root, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.368 NppStatus nppsSqrt_16sc_ISfs (Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit complex signed short signal square root, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.369 NppStatus nppsSqrt_16sc_Sfs (const Npp16sc * *pSrc*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit complex signed short signal square root, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.370 NppStatus nppsSqrt_16u_ISfs (Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short signal square root, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.371 NppStatus nppsSqrt_16u_Sfs (const Npp16u * *pSrc*, Npp16u * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short signal square root, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.372 NppStatus nppsSqrt_32f (const Npp32f * pSrc, Npp32f * pDst, int nLength)

32-bit floating point signal square root.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.373 NppStatus nppsSqrt_32f_I (Npp32f * pSrcDst, int nLength)

32-bit floating point signal square root.

Parameters:

pSrcDst In-Place Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.374 NppStatus nppsSqrt_32fc (const Npp32fc * pSrc, Npp32fc * pDst, int nLength)

32-bit complex floating point signal square root.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.375 NppStatus nppsSqrt_32fc_I (Npp32fc * *pSrcDst*, int *nLength*)

32-bit complex floating point signal square root.

Parameters:

pSrcDst In-Place Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.376 NppStatus nppsSqrt_32s16s_Sfs (const Npp32s * *pSrc*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal square root, scale, then clamp to 16-bit signed integer saturated value.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.377 NppStatus nppsSqrt_64f (const Npp64f * *pSrc*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal square root.

Parameters:

pSrc Source Signal Pointer.
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.378 NppStatus nppsSqrt_64f_I (Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point signal square root.

Parameters:

pSrcDst In-Place Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.379 NppStatus nppsSqrt_64fc (const Npp64fc * *pSrc*, Npp64fc * *pDst*, int *nLength*)

64-bit complex floating point signal square root.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.380 NppStatus nppsSqrt_64fc_I (Npp64fc * *pSrcDst*, int *nLength*)

64-bit complex floating point signal square root.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.381 NppStatus nppsSqrt_64s16s_Sfs (const Npp64s * *pSrc*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

64-bit signed integer signal square root, scale, then clamp to 16-bit signed integer saturated value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.382 NppStatus nppsSqrt_64s_ISfs (Npp64s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

64-bit signed integer signal square root, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.383 NppStatus nppsSqrt_64s_Sfs (const Npp64s * *pSrc*, Npp64s * *pDst*, int *nLength*, int *nScaleFactor*)

64-bit signed integer signal square root, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.384 NppStatus nppsSqrt_8u_ISfs (Npp8u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char signal square root, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.385 NppStatus nppsSqrt_8u_Sfs (const Npp8u * *pSrc*, Npp8u * *pDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char signal square root, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.386 **NppStatus nppsSub_16s** (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp16s * *pDst*, int *nLength*)

16-bit signed short signal subtract signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.387 **NppStatus nppsSub_16s32f** (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp32f * *pDst*, int *nLength*)

16-bit signed short signal subtract 16-bit signed short signal, then clamp and convert to 32-bit floating point saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.388 **NppStatus nppsSub_16s_I** (const Npp16s * *pSrc*, Npp16s * *pSrcDst*, int *nLength*)

16-bit signed short in place signal subtract signal, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.389 NppStatus nppsSub_16s_ISfs (const Npp16s * *pSrc*, Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short in place signal subtract signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal1 elements to be subtracted from signal2 elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.390 NppStatus nppsSub_16s_Sfs (const Npp16s * *pSrc1*, const Npp16s * *pSrc2*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal subtract signal, scale, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer, signal1 elements to be subtracted from signal2 elements.

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.391 NppStatus nppsSub_16sc_ISfs (const Npp16sc * *pSrc*, Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit complex signed short in place signal subtract signal, with scaling, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal1 elements to be subtracted from signal2 elements

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.392 NppStatus nppsSub_16sc_Sfs (const Npp16sc * *pSrc1*, const Npp16sc * *pSrc2*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed complex short signal subtract signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal1 elements to be subtracted from signal2 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.393 NppStatus nppsSub_16u_ISfs (const Npp16u * *pSrc*, Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short in place signal subtract signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.394 NppStatus nppsSub_16u_Sfs (const Npp16u * *pSrc1*, const Npp16u * *pSrc2*, Npp16u * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short signal subtract signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal1 elements to be subtracted from signal2 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.395 NppStatus nppsSub_32f (const Npp32f * *pSrc1*, const Npp32f * *pSrc2*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal subtract signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal1 elements to be subtracted from signal2 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.396 NppStatus nppsSub_32f_I (const Npp32f * *pSrc*, Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point in place signal subtract signal, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

pSrcDst In-Place Signal Pointer. signal1 elements to be subtracted from signal2 elements

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.397 NppStatus nppsSub_32fc (const Npp32fc * *pSrc1*, const Npp32fc * *pSrc2*, Npp32fc * *pDst*, int *nLength*)

32-bit complex floating point signal subtract signal, then clamp to saturated value.

Parameters:

pSrc1 Source Signal Pointer.

pSrc2 Source Signal Pointer. signal1 elements to be subtracted from signal2 elements

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.398 NppStatus nppsSub_32fc_I (const Npp32fc * *pSrc*, Npp32fc * *pSrcDst*, int *nLength*)

32-bit complex floating point in place signal subtract signal, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.399 NppStatus nppsSub_32s_ISfs (const Npp32s * *pSrc*, Npp32s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer in place signal subtract signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.400 NppStatus nppsSub_32s_Sfs (const Npp32s * *pSrc1*, const Npp32s * *pSrc2*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer signal subtract signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal1 elements to be subtracted from signal2 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.401 NppStatus nppsSub_32sc_ISfs (const Npp32sc * *pSrc*, Npp32sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit complex signed integer in place signal subtract signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.402 NppStatus nppsSub_32sc_Sfs (const Npp32sc * *pSrc1*, const Npp32sc * *pSrc2*, Npp32sc * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed complex integer signal subtract signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal1 elements to be subtracted from signal2 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.403 NppStatus nppsSub_64f (const Npp64f * *pSrc1*, const Npp64f * *pSrc2*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal subtract signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.404 NppStatus nppsSub_64f_I (const Npp64f * *pSrc*, Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point in place signal subtract signal, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.405 NppStatus nppsSub_64fc (const Npp64fc * *pSrc1*, const Npp64fc * *pSrc2*, Npp64fc * *pDst*, int *nLength*)

64-bit complex floating point signal subtract signal, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.406 NppStatus nppsSub_64fc_I (const Npp64fc * *pSrc*, Npp64fc * *pSrcDst*, int *nLength*)

64-bit complex floating point in place signal subtract signal, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.407 NppStatus nppsSub_8u_ISfs (const Npp8u * *pSrc*, Npp8u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char in place signal subtract signal, with scaling, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal1 elements to be subtracted from signal2 elements

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.408 `NppStatus nppsSub_8u_Sfs (const Npp8u * pSrc1, const Npp8u * pSrc2, Npp8u * pDst, int nLength, int nScaleFactor)`

8-bit unsigned char signal subtract signal, scale, then clamp to saturated value.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#), signal1 elements to be subtracted from signal2 elements.

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.409 `NppStatus nppsSubC_16s_ISfs (Npp16s nValue, Npp16s * pSrcDst, int nLength, int nScaleFactor)`

16-bit signed short in place signal subtract constant, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.410 **NppStatus nppsSubC_16s_Sfs** (const Npp16s * *pSrc*, Npp16s *nValue*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.411 **NppStatus nppsSubC_16sc_ISfs** (Npp16sc *nValue*, Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal subtract constant, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.412 **NppStatus nppsSubC_16sc_Sfs** (const Npp16sc * *pSrc*, Npp16sc *nValue*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.413 NppStatus nppsSubC_16u_ISfs (Npp16u *nValue*, Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short in place signal subtract constant, scale, then clamp to saturated value.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be subtracted from each vector element

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.414 NppStatus nppsSubC_16u_Sfs (const Npp16u * *pSrc*, Npp16u *nValue*, Npp16u * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short signal subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be subtracted from each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.415 NppStatus nppsSubC_32f (const Npp32f * *pSrc*, Npp32f *nValue*, Npp32f * *pDst*, int *nLength*)

32-bit floating point signal subtract constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be subtracted from each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.416 NppStatus nppsSubC_32f_I (Npp32f *nValue*, Npp32f * *pSrcDst*, int *nLength*)

32-bit floating point in place signal subtract constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.417 NppStatus nppsSubC_32fc (const Npp32fc * *pSrc*, Npp32fc *nValue*, Npp32fc * *pDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) signal subtract constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.418 NppStatus nppsSubC_32fc_I (Npp32fc *nValue*, Npp32fc * *pSrcDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal subtract constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.419 NppStatus nppsSubC_32s_ISfs (Npp32s *nValue*, Npp32s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer in place signal subtract constant and scale.

Parameters:

pSrcDst In-Place Signal Pointer.
nValue Constant value to be subtracted from each vector element
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.420 `NppStatus nppsSubC_32s_Sfs (const Npp32s * pSrc, Npp32s nValue, Npp32s * pDst, int nLength, int nScaleFactor)`

32-bit signed integer signal subtract constant and scale.

Parameters:

pSrc Source Signal Pointer.
nValue Constant value to be subtracted from each vector element
pDst Destination Signal Pointer.
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.421 `NppStatus nppsSubC_32sc_ISfs (Npp32sc nValue, Npp32sc * pSrcDst, int nLength, int nScaleFactor)`

32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal subtract constant and scale.

Parameters:

pSrcDst In-Place Signal Pointer.
nValue Constant value to be subtracted from each vector element
nLength Signal Length.
nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.422 NppStatus nppsSubC_32sc_Sfs (const Npp32sc * *pSrc*, Npp32sc *nValue*, Npp32sc * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) signal subtract constant and scale.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.423 NppStatus nppsSubC_64f (const Npp64f * *pSrc*, Npp64f *nValue*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal subtract constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.424 NppStatus nppsSubC_64f_I (Npp64f *nValue*, Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point, in place signal subtract constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

nLength Length of the vectors, number of items.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.425 NppStatus nppsSubC_64fc (const Npp64fc * *pSrc*, Npp64fc *nValue*, Npp64fc * *pDst*, int *nLength*)

64-bit floating point complex number (64 bit real, 64 bit imaginary) signal subtract constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be subtracted from each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.426 NppStatus nppsSubC_64fc_I (Npp64fc *nValue*, Npp64fc * *pSrcDst*, int *nLength*)

64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal subtract constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be subtracted from each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.427 NppStatus nppsSubC_8u_ISfs (Npp8u *nValue*, Npp8u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char in place signal subtract constant, scale, then clamp to saturated value

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be subtracted from each vector element

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.428 NppStatus nppsSubC_8u_Sfs (const Npp8u * *pSrc*, Npp8u *nValue*, Npp8u * *pDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char signal subtract constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value to be subtracted from each vector element

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.429 NppStatus nppsSubCRev_16s_ISfs (Npp16s *nValue*, Npp16s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short in place signal subtract from constant, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.430 NppStatus nppsSubCRev_16s_Sfs (const Npp16s * *pSrc*, Npp16s *nValue*, Npp16s * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit signed short signal subtract from constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.431 NppStatus nppsSubCRev_16sc_ISfs (Npp16sc *nValue*, Npp16sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal subtract from constant, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.432 NppStatus nppsSubCRev_16sc_Sfs (const Npp16sc * *pSrc*, Npp16sc *nValue*, Npp16sc * *pDst*, int *nLength*, int *nScaleFactor*)

16-bit integer complex number (16 bit real, 16 bit imaginary) signal subtract from constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.433 NppStatus nppsSubCRev_16u_ISfs (Npp16u *nValue*, Npp16u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

16-bit unsigned short in place signal subtract from constant, scale, then clamp to saturated value.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.434 `NppStatus nppsSubCRev_16u_Sfs (const Npp16u * pSrc, Npp16u nValue, Npp16u * pDst, int nLength, int nScaleFactor)`

16-bit unsigned short signal subtract from constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.435 `NppStatus nppsSubCRev_32f (const Npp32f * pSrc, Npp32f nValue, Npp32f * pDst, int nLength)`

32-bit floating point signal subtract from constant.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.436 `NppStatus nppsSubCRev_32f_I (Npp32f nValue, Npp32f * pSrcDst, int nLength)`

32-bit floating point in place signal subtract from constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.437 NppStatus nppsSubCRev_32fc (const Npp32fc * *pSrc*, Npp32fc *nValue*, Npp32fc * *pDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) signal subtract from constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value each vector element is to be subtracted from

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.438 NppStatus nppsSubCRev_32fc_I (Npp32fc *nValue*, Npp32fc * *pSrcDst*, int *nLength*)

32-bit floating point complex number (32 bit real, 32 bit imaginary) in place signal subtract from constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value each vector element is to be subtracted from

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.439 NppStatus nppsSubCRev_32s_ISfs (Npp32s *nValue*, Npp32s * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integer in place signal subtract from constant and scale.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value each vector element is to be subtracted from

nLength Signal Length.

nScaleFactor Integer Result Scaling.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.440 NppStatus nppsSubCRev_32s_Sfs (const Npp32s * *pSrc*, Npp32s *nValue*, Npp32s * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit signed integersignal subtract from constant and scale.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.441 NppStatus nppsSubCRev_32sc_ISfs (Npp32sc *nValue*, Npp32sc * *pSrcDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) in place signal subtract from constant and scale.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.442 NppStatus nppsSubCRev_32sc_Sfs (const Npp32sc * *pSrc*, Npp32sc *nValue*, Npp32sc * *pDst*, int *nLength*, int *nScaleFactor*)

32-bit integer complex number (32 bit real, 32 bit imaginary) signal subtract from constant and scale.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.443 NppStatus nppsSubCRev_64f (const Npp64f * *pSrc*, Npp64f *nValue*, Npp64f * *pDst*, int *nLength*)

64-bit floating point signal subtract from constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value each vector element is to be subtracted from

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.444 NppStatus nppsSubCRev_64f_I (Npp64f *nValue*, Npp64f * *pSrcDst*, int *nLength*)

64-bit floating point, in place signal subtract from constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value each vector element is to be subtracted from

nLength Length of the vectors, number of items.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.445 NppStatus nppsSubCRev_64fc (const Npp64fc * *pSrc*, Npp64fc *nValue*, Npp64fc * *pDst*, int *nLength*)

64-bit floating point complex number (64 bit real, 64 bit imaginary) signal subtract from constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value each vector element is to be subtracted from

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.446 NppStatus nppsSubCRev_64fc_I (Npp64fc *nValue*, Npp64fc * *pSrcDst*, int *nLength*)

64-bit floating point complex number (64 bit real, 64 bit imaginary) in place signal subtract from constant.

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.447 NppStatus nppsSubCRev_8u_ISfs (Npp8u *nValue*, Npp8u * *pSrcDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char in place signal subtract from constant, scale, then clamp to saturated value

Parameters:

pSrcDst [In-Place Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.448 NppStatus nppsSubCRev_8u_Sfs (const Npp8u * *pSrc*, Npp8u *nValue*, Npp8u * *pDst*, int *nLength*, int *nScaleFactor*)

8-bit unsigned char signal subtract from constant, scale, then clamp to saturated value.

Parameters:

pSrc [Source Signal Pointer](#).

nValue Constant value each vector element is to be subtracted from

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

nScaleFactor [Integer Result Scaling](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.449 NppStatus nppsSum_16s32s_Sfs (const Npp16s * pSrc, int nLength, Npp32s * pSum, int nScaleFactor, Npp8u * pDeviceBuffer)

16-bit integer vector sum (32bit) with integer scaling method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pSum Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

nScaleFactor Integer-result scale factor.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.450 NppStatus nppsSum_16s_Sfs (const Npp16s * pSrc, int nLength, Npp16s * pSum, int nScaleFactor, Npp8u * pDeviceBuffer)

16-bit short vector sum with integer scaling method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pSum Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

nScaleFactor Integer-result scale factor.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.451 NppStatus nppsSum_16sc32sc_Sfs (const Npp16sc * pSrc, int nLength, Npp32sc * pSum, int nScaleFactor, Npp8u * pDeviceBuffer)

16-bit short complex vector sum (32bit int complex) with integer scaling method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pSum Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

nScaleFactor Integer-result scale factor.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.452 NppStatus nppsSum_16sc_Sfs (const Npp16sc * *pSrc*, int *nLength*, Npp16sc * *pSum*, int *nScaleFactor*, Npp8u * *pDeviceBuffer*)

16-bit short complex vector sum with integer scaling method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pSum Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

nScaleFactor Integer-result scale factor.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.453 NppStatus nppsSum_32f (const Npp32f * *pSrc*, int *nLength*, Npp32f * *pSum*, NppHintAlgorithm *eHint*, Npp8u * *pDeviceBuffer*)

32-bit float vector sum method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pSum Pointer to the output result.

eHint Suggests using specific code.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.454 NppStatus nppsSum_32fc (const Npp32fc * *pSrc*, int *nLength*, Npp32fc * *pSum*, NppHintAlgorithm *eHint*, Npp8u * *pDeviceBuffer*)

32-bit float complex vector sum method

Parameters:

pSrc [Source Signal Pointer](#).

nLength [Signal Length](#).

pSum Pointer to the output result.

eHint Suggests using specific code.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.455 `NppStatus nppsSum_32s_Sfs (const Npp32s * pSrc, int nLength, Npp32s * pSum, int nScaleFactor, Npp8u * pDeviceBuffer)`

32-bit integer vector sum with integer scaling method

Parameters:

pSrc Source Signal Pointer.

nLength Signal Length.

pSum Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

nScaleFactor Integer-result scale factor.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.456 `NppStatus nppsSum_64f (const Npp64f * pSrc, int nLength, Npp64f * pSum, Npp8u * pDeviceBuffer)`

64-bit double vector sum method

Parameters:

pSrc Source Signal Pointer.

nLength Signal Length.

pSum Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.457 `NppStatus nppsSum_64fc (const Npp64fc * pSrc, int nLength, Npp64fc * pSum, Npp8u * pDeviceBuffer)`

64-bit double complex vector sum method

Parameters:

pSrc Source Signal Pointer.

nLength Signal Length.

pSum Pointer to the output result.

pDeviceBuffer Pointer to the required device memory allocation.

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.458 NppStatus nppsSumGetBufferSize_16s32s_Sfs (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 16s32s_Sfs.

This primitive provides the correct buffer size for nppsSum_16s32s_Sfs.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.459 NppStatus nppsSumGetBufferSize_16s_Sfs (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 16s_Sfs Sum.

This primitive provides the correct buffer size for nppsSum_16s_Sfs.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.460 NppStatus nppsSumGetBufferSize_16sc32sc_Sfs (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 16sc32sc_Sfs Sum.

This primitive provides the correct buffer size for nppsSum_16sc32sc_Sfs.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.461 NppStatus nppsSumGetBufferSize_16sc_Sfs (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 16sc_Sfs Sum.

This primitive provides the correct buffer size for nppsSum_16sc_Sfs.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.462 NppStatus nppsSumGetBufferSize_32f (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 32f Sum.

This primitive provides the correct buffer size for `nppsSum_32f`.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.463 NppStatus nppsSumGetBufferSize_32fc (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 32fc Sum.

This primitive provides the correct buffer size for `nppsSum_32fc`.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.464 NppStatus nppsSumGetBufferSize_32s_Sfs (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 32s_Sfs Sum.

This primitive provides the correct buffer size for `nppsSum_32s_Sfs`.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.465 NppStatus nppsSumGetBufferSize_64f (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 64f Sum.

This primitive provides the correct buffer size for nppsSum_64f.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.466 NppStatus nppsSumGetBufferSize_64fc (int *nLength*, int * *hpBufferSize*)

Device scratch buffer size (in bytes) for 64fc Sum.

This primitive provides the correct buffer size for nppsSum_64fc.

Parameters:

nLength [Signal Length](#).

hpBufferSize Required buffer size. Important: *hpBufferSize* is a *host pointer*. [Scratch Buffer and Host Pointer](#).

Returns:

NPP_SUCCESS

7.17.1.467 NppStatus nppsXor_16u (const Npp16u * *pSrc1*, const Npp16u * *pSrc2*, Npp16u * *pDst*, int *nLength*)

16-bit unsigned short signal exclusive or with signal.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal2 elements to be exclusive ored with signal1 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.468 NppStatus nppsXor_16u_I (const Npp16u * *pSrc*, Npp16u * *pSrcDst*, int *nLength*)

16-bit unsigned short in place signal exclusive or with signal.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be exclusive ored with signal1 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.469 NppStatus nppsXor_32u (const Npp32u * *pSrc1*, const Npp32u * *pSrc2*, Npp32u * *pDst*, int *nLength*)

32-bit unsigned integer signal exclusive or with signal.

Parameters:

pSrc1 [Source Signal Pointer](#).

pSrc2 [Source Signal Pointer](#). signal2 elements to be exclusive ored with signal1 elements

pDst [Destination Signal Pointer](#).

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.470 NppStatus nppsXor_32u_I (const Npp32u * *pSrc*, Npp32u * *pSrcDst*, int *nLength*)

32-bit unsigned integer in place signal exclusive or with signal.

Parameters:

pSrc [Source Signal Pointer](#).

pSrcDst [In-Place Signal Pointer](#). signal2 elements to be exclusive ored with signal1 elements

nLength [Signal Length](#).

Returns:

[Signal Data Related Error Codes](#), [Length Related Error Codes](#).

7.17.1.471 NppStatus nppsXor_8u (const Npp8u * *pSrc1*, const Npp8u * *pSrc2*, Npp8u * *pDst*, int *nLength*)

8-bit unsigned char signal exclusive or with signal.

Parameters:

pSrc1 Source Signal Pointer.
pSrc2 Source Signal Pointer. signal2 elements to be exclusive ored with signal1 elements
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.472 NppStatus nppsXor_8u_I (const Npp8u * pSrc, Npp8u * pSrcDst, int nLength)

8-bit unsigned char in place signal exclusive or with signal.

Parameters:

pSrc Source Signal Pointer.
pSrcDst In-Place Signal Pointer. signal2 elements to be exclusive ored with signal1 elements
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.473 NppStatus nppsXorC_16u (const Npp16u * pSrc, Npp16u nValue, Npp16u * pDst, int nLength)

16-bit unsigned short signal exclusive or with constant.

Parameters:

pSrc Source Signal Pointer.
nValue Constant value to be exclusive ored with each vector element
pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.474 NppStatus nppsXorC_16u_I (Npp16u nValue, Npp16u * pSrcDst, int nLength)

16-bit unsigned short in place signal exclusive or with constant.

Parameters:

pSrcDst In-Place Signal Pointer.
nValue Constant value to be exclusive ored with each vector element
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.475 NppStatus nppsXorC_32u (const Npp32u * *pSrc*, Npp32u *nValue*, Npp32u * *pDst*, int *nLength*)

32-bit unsigned integer signal exclusive or with constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be exclusive ored with each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.476 NppStatus nppsXorC_32u_I (Npp32u *nValue*, Npp32u * *pSrcDst*, int *nLength*)

32-bit unsigned signed integer in place signal exclusive or with constant.

Parameters:

pSrcDst In-Place Signal Pointer.

nValue Constant value to be exclusive ored with each vector element

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.477 NppStatus nppsXorC_8u (const Npp8u * *pSrc*, Npp8u *nValue*, Npp8u * *pDst*, int *nLength*)

8-bit unsigned char signal exclusive or with constant.

Parameters:

pSrc Source Signal Pointer.

nValue Constant value to be exclusive ored with each vector element

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.478 NppStatus nppsXorC_8u_I (Npp8u *nValue*, Npp8u * *pSrcDst*, int *nLength*)

8-bit unsigned char in place signal exclusive or with constant.

Parameters:

pSrcDst In-Place Signal Pointer.
nValue Constant value to be exclusive ored with each vector element
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.479 NppStatus nppsZero_16s (Npp16s * *pDst*, int *nLength*)

16-bit integer, vector zero method.

Parameters:

pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.480 NppStatus nppsZero_16sc (Npp16sc * *pDst*, int *nLength*)

16-bit integer complex, vector zero method.

Parameters:

pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.481 NppStatus nppsZero_32f (Npp32f * *pDst*, int *nLength*)

32-bit float, vector zero method.

Parameters:

pDst Destination Signal Pointer.
nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.482 NppStatus nppsZero_32fc (Npp32fc * *pDst*, int *nLength*)

32-bit float complex, vector zero method.

Parameters:

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.483 NppStatus nppsZero_32s (Npp32s * *pDst*, int *nLength*)

32-bit integer, vector zero method.

Parameters:

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.484 NppStatus nppsZero_32sc (Npp32sc * *pDst*, int *nLength*)

32-bit integer complex, vector zero method.

Parameters:

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.485 NppStatus nppsZero_64f (Npp64f * *pDst*, int *nLength*)

64-bit double, vector zero method.

Parameters:

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.486 NppStatus nppsZero_64fc (Npp64fc * *pDst*, int *nLength*)

64-bit double complex, vector zero method.

Parameters:

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.487 NppStatus nppsZero_64s (Npp64s * *pDst*, int *nLength*)

64-bit long long integer, vector zero method.

Parameters:

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.488 NppStatus nppsZero_64sc (Npp64sc * *pDst*, int *nLength*)

64-bit long long integer complex, vector zero method.

Parameters:

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

7.17.1.489 NppStatus nppsZero_8u (Npp8u * *pDst*, int *nLength*)

8-bit unsigned char, vector zero method.

Parameters:

pDst Destination Signal Pointer.

nLength Signal Length.

Returns:

Signal Data Related Error Codes, Length Related Error Codes.

Chapter 8

Data Structure Documentation

8.1 Npp16sc Struct Reference

Complex Number This struct represents a short complex number.

```
#include <nppdefs.h>
```

Data Fields

- [Npp16s re](#)
Real part.
- [Npp16s im](#)
Imaginary part.

8.1.1 Detailed Description

Complex Number This struct represents a short complex number.

8.1.2 Field Documentation

8.1.2.1 Npp16s Npp16sc::im

Imaginary part.

8.1.2.2 Npp16s Npp16sc::re

Real part.

The documentation for this struct was generated from the following file:

- C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h

8.2 Npp32fc Struct Reference

Complex Number This struct represents a single floating-point complex number.

```
#include <nppdefs.h>
```

Data Fields

- [Npp32f re](#)
Real part.
- [Npp32f im](#)
Imaginary part.

8.2.1 Detailed Description

Complex Number This struct represents a single floating-point complex number.

8.2.2 Field Documentation

8.2.2.1 Npp32f Npp32fc::im

Imaginary part.

8.2.2.2 Npp32f Npp32fc::re

Real part.

The documentation for this struct was generated from the following file:

- C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h

8.3 Npp32sc Struct Reference

Complex Number This struct represents a signed int complex number.

```
#include <nppdefs.h>
```

Data Fields

- [Npp32s re](#)
Real part.
- [Npp32s im](#)
Imaginary part.

8.3.1 Detailed Description

Complex Number This struct represents a signed int complex number.

8.3.2 Field Documentation

8.3.2.1 Npp32s Npp32sc::im

Imaginary part.

8.3.2.2 Npp32s Npp32sc::re

Real part.

The documentation for this struct was generated from the following file:

- C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h

8.4 Npp64fc Struct Reference

Complex Number This struct represents a double floating-point complex number.

```
#include <nppdefs.h>
```

Data Fields

- [Npp64f re](#)
Real part.
- [Npp64f im](#)
Imaginary part.

8.4.1 Detailed Description

Complex Number This struct represents a double floating-point complex number.

8.4.2 Field Documentation

8.4.2.1 Npp64f Npp64fc::im

Imaginary part.

8.4.2.2 Npp64f Npp64fc::re

Real part.

The documentation for this struct was generated from the following file:

- C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h

8.5 Npp64sc Struct Reference

Complex Number This struct represents a long long complex number.

```
#include <nppdefs.h>
```

Data Fields

- [Npp64s re](#)
Real part.
- [Npp64s im](#)
Imaginary part.

8.5.1 Detailed Description

Complex Number This struct represents a long long complex number.

8.5.2 Field Documentation

8.5.2.1 Npp64s Npp64sc::im

Imaginary part.

8.5.2.2 Npp64s Npp64sc::re

Real part.

The documentation for this struct was generated from the following file:

- C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h

8.6 NppiHaarBuffer Struct Reference

```
#include <nppdefs.h>
```

Data Fields

- `int haarBufferSize`
size of the buffer
- `Npp32s * haarBuffer`
buffer

8.6.1 Field Documentation

8.6.1.1 `Npp32s* NppiHaarBuffer::haarBuffer`

buffer

8.6.1.2 `int NppiHaarBuffer::haarBufferSize`

size of the buffer

The documentation for this struct was generated from the following file:

- `C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h`

8.7 NppiHaarClassifier_32f Struct Reference

```
#include <nppdefs.h>
```

Data Fields

- int [numClassifiers](#)
number of classifiers
- [Npp32s](#) * [classifiers](#)
packed classifier data 40 bytes each
- [size_t](#) [classifierStep](#)
- [NppiSize](#) [classifierSize](#)
- [Npp32s](#) * [counterDevice](#)

8.7.1 Field Documentation

8.7.1.1 [Npp32s](#)* [NppiHaarClassifier_32f::classifiers](#)

packed classifier data 40 bytes each

8.7.1.2 [NppiSize](#) [NppiHaarClassifier_32f::classifierSize](#)

8.7.1.3 [size_t](#) [NppiHaarClassifier_32f::classifierStep](#)

8.7.1.4 [Npp32s](#)* [NppiHaarClassifier_32f::counterDevice](#)

8.7.1.5 [int](#) [NppiHaarClassifier_32f::numClassifiers](#)

number of classifiers

The documentation for this struct was generated from the following file:

- C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h

8.8 NppiPoint Struct Reference

2D Point

```
#include <nppdefs.h>
```

Data Fields

- `int x`
x-coordinate.
- `int y`
y-coordinate.

8.8.1 Detailed Description

2D Point

8.8.2 Field Documentation

8.8.2.1 `int NppiPoint::x`

x-coordinate.

8.8.2.2 `int NppiPoint::y`

y-coordinate.

The documentation for this struct was generated from the following file:

- `C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h`

8.9 NppiRect Struct Reference

2D Rectangle This struct contains position and size information of a rectangle in two space.

```
#include <nppdefs.h>
```

Data Fields

- `int x`
x-coordinate of upper left corner.
- `int y`
y-coordinate of upper left corner.
- `int width`
Rectangle width.
- `int height`
Rectangle height.

8.9.1 Detailed Description

2D Rectangle This struct contains position and size information of a rectangle in two space.

The rectangle's position is usually signified by the coordinate of its upper-left corner.

8.9.2 Field Documentation

8.9.2.1 `int NppiRect::height`

Rectangle height.

8.9.2.2 `int NppiRect::width`

Rectangle width.

8.9.2.3 `int NppiRect::x`

x-coordinate of upper left corner.

8.9.2.4 `int NppiRect::y`

y-coordinate of upper left corner.

The documentation for this struct was generated from the following file:

- `C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h`

8.10 NppiSize Struct Reference

2D Size This struct typically represents the size of a rectangular region in two space.

```
#include <nppdefs.h>
```

Data Fields

- int [width](#)
Rectangle width.
- int [height](#)
Rectangle height.

8.10.1 Detailed Description

2D Size This struct typically represents the size of a rectangular region in two space.

8.10.2 Field Documentation

8.10.2.1 int NppiSize::height

Rectangle height.

8.10.2.2 int NppiSize::width

Rectangle width.

The documentation for this struct was generated from the following file:

- C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h

8.11 NppLibraryVersion Struct Reference

```
#include <nppdefs.h>
```

Data Fields

- int `major`
Major version number.
- int `minor`
Minor version number.
- int `build`
Build number. This reflects the nightly build this release was made from.

8.11.1 Field Documentation

8.11.1.1 int NppLibraryVersion::build

Build number. This reflects the nightly build this release was made from.

8.11.1.2 int NppLibraryVersion::major

Major version number.

8.11.1.3 int NppLibraryVersion::minor

Minor version number.

The documentation for this struct was generated from the following file:

- C:/Perforce/sw/rel/gpgpu/toolkit/r4.1/NPP/npp/include/nppdefs.h