

# TeliCamDNetAPI

---

## Library manual

Version 2.0.3 (2017/09/05)

**TOSHIBA TELI CORPORATION**

Information contained in this document is subject to change without prior notice.

---

## Contents

1. Introduction.....	11
2. Configuration .....	12
3. Operation Environment .....	13
4. Library .....	14
4.1. Wrapping TeliCamAPI functions.....	14
4.2. Difference of usage between TeliCamAPI functions .....	16
4.3. Installing SDK.....	18
4.4. Uninstalling SDK .....	18
4.5. Developing .NET Framework application.....	18
5. Classes in TeliCamDNetAPI.....	19
5.1. CameraSystem class .....	20
5.1.1. Constructors .....	21
5.1.2. Methods.....	22
5.1.2.1. Dispose method.....	22
5.1.2.2. Initialize method.....	23
5.1.2.3. Terminate method .....	24
5.1.2.4. GetInformation method.....	25
5.1.2.5. GetNumOfCameras method .....	26
5.1.2.6. GetCameraInformation method .....	27
5.1.2.7. CreateDeviceObject method .....	28
5.1.2.8. CreateDeviceObjectFromInfo method .....	29
5.2. CameraDevice class .....	30
5.2.1. Methods.....	32
5.2.1.1. Dispose method.....	32
5.2.1.2. GetInformation method .....	33
5.2.1.3. Open method .....	34
5.2.1.4. Close method.....	36
5.2.1.5. ReadRegister method.....	37
5.2.1.6. WriteRegister method .....	39
5.2.1.7. ResetPort method .....	41
5.2.1.8. GetHeartbeat method .....	42
5.2.1.9. SetHeartbeat method.....	43
5.2.1.10. GetTLParamsLocked method.....	44
5.2.1.11. GetLastGenICamError method.....	45
5.2.2. Member Variables (Fields) .....	46
5.2.2.1. camStream field .....	46
5.2.2.2. camEvent field .....	47
5.2.2.3. camControl field .....	48
5.2.2.4. genApi field .....	49
5.2.3. Properties .....	50
5.2.3.1. CamType property .....	50
5.2.3.2. CamIndex property .....	51
5.2.3.3. IsOpen property .....	52
5.2.4. Events.....	53
5.2.4.1. Removed event.....	53
5.3. CameraStream class.....	54
5.3.1. Methods.....	56

5.3.1.1.	Open method .....	56
5.3.1.2.	Close method.....	59
5.3.1.3.	Start method .....	60
5.3.1.4.	Stop method.....	62
5.3.1.5.	Abort method .....	63
5.3.1.6.	ReadCurrentImage methos .....	64
5.3.1.7.	GetCurrentBufferIndex method.....	66
5.3.1.8.	LockBuffer method.....	67
5.3.1.9.	UnlockBuffer method .....	68
5.3.1.10.	ChunkAttachBuffer method.....	69
5.3.2.	Properties .....	71
5.3.2.1.	Parent property .....	71
5.3.3.	Events.....	72
5.3.3.1.	ImageAcquired event.....	72
5.3.3.2.	ImageErrorReceived event.....	73
5.3.3.3.	BufferBusyReceived event .....	74
5.4.	CameraEvent class .....	75
5.4.1.	Methods.....	77
5.4.1.1.	Open method .....	77
5.4.1.2.	Close method.....	79
5.4.1.3.	Activate method .....	80
5.4.1.4.	Deactivate method.....	83
5.4.2.	Properties .....	84
5.4.2.1.	Parent property .....	84
5.4.3.	Events.....	85
5.4.3.1.	FrameTriggerReceived event .....	85
5.4.3.2.	FrameTriggerErrorReceived event .....	85
5.4.3.3.	FrameTriggerWaitReceived event .....	86
5.4.3.4.	FrameTransferStartReceived event.....	86
5.4.3.5.	FrameTransferEndReceived event.....	87
5.4.3.6.	ExposureStartReceived event .....	87
5.4.3.7.	ExposureEndReceived event .....	88
5.4.3.8.	Timer0StartReceived event .....	88
5.4.3.9.	Timer0EndReceived event .....	89
5.4.3.10.	ALCLatestInformationReceived event .....	89
5.4.3.11.	ALCConvergedReceived event .....	90
5.5.	CameraControl class.....	91
5.5.1.	ImageFormatControl methods.....	97
5.5.1.1.	GetImageFormatSelector method .....	97
5.5.1.2.	SetImageFormatSelector method.....	98
5.5.2.	Scalable methods .....	99
5.5.2.1.	GetSensorWidth method .....	99
5.5.2.2.	GetSensorHeight method .....	100
5.5.2.3.	GetRoi method.....	101
5.5.2.4.	SetRoi method .....	102
5.5.2.5.	GetWidthMinMax method .....	103
5.5.2.6.	GetWidth method.....	104
5.5.2.7.	SetWidth method .....	105
5.5.2.8.	GetHeightMinMax method .....	106

---

5.5.2.9. GetHeight method .....	107
5.5.2.10. SetHeight method .....	108
5.5.2.11. GetOffsetXMinMax method .....	109
5.5.2.12. GetOffsetX method .....	110
5.5.2.13. SetOffsetX method .....	111
5.5.2.14. GetOffsetYMinMax method .....	112
5.5.2.15. GetOffsetY method .....	113
5.5.2.16. SetOffsetY method.....	114
5.5.3. Binning methods.....	115
5.5.3.1. GetBinningHorizontalMinMax method .....	115
5.5.3.2. GetBinningHorizontal method.....	116
5.5.3.3. SetBinningHorizontal method .....	117
5.5.3.4. GetBinningVerticalMinMax method .....	118
5.5.3.5. GetBinningVertical method .....	119
5.5.3.6. SetBinningVertical method .....	120
5.5.4. Decimation methods.....	121
5.5.4.1. GetDecimationHorizontalMinMax method .....	121
5.5.4.2. GetDecimationHorizontal method.....	122
5.5.4.3. SetDecimationHorizontal method .....	123
5.5.4.4. GetDecimationVerticalMinMax method .....	124
5.5.4.5. GetDecimationVertical method .....	125
5.5.4.6. SetDecimationVertical method .....	126
5.5.5. Reverse methods .....	127
5.5.5.1. GetReverseX method .....	127
5.5.5.2. SetReverseX method.....	128
5.5.5.3. GetReverseY method .....	129
5.5.5.4. SetReverseY method.....	130
5.5.6. PixelFormat methods .....	131
5.5.6.1. GetPixelFormat method.....	131
5.5.6.2. SetPixelFormat method .....	132
5.5.7. TestPattern methods .....	133
5.5.7.1. GetTestPattern method.....	133
5.5.7.2. SetTestPattern method .....	134
5.5.8. AcquisitionControl methods.....	135
5.5.8.1. GetStreamPayloadSize mesod.....	135
5.5.8.2. GetStreamEnable method .....	136
5.5.8.3. GetAcquisitionFrameCountMinMax method.....	137
5.5.8.4. GetAcquisitionFrameCount method .....	138
5.5.8.5. SetAcquisitionFrameCount method.....	139
5.5.8.6. GetAcquisitionFrameRateControl method.....	140
5.5.8.7. SetAcquisitionFrameRateControl method .....	141
5.5.8.8. GetAcquisitionFrameRateMinMax method.....	142
5.5.8.9. GetAcquisitionFrameRate method .....	143
5.5.8.10. SetAcquisitionFrameRate method.....	144
5.5.9. ImageBuffer methods .....	145
5.5.9.1. GetImageBufferMode method .....	145
5.5.9.2. SetImageBufferMode mode.....	146
5.5.9.3. GetImageBufferFrameCount method .....	147
5.5.9.4. ExecutImageBufferRead method .....	148

---

---

5.5.10. TriggerControl methods.....	149
5.5.10.1. GetTriggerMode method.....	149
5.5.10.2. SetTriggerMode method.....	150
5.5.10.3. GetTriggerSequence method.....	151
5.5.10.4. SetTriggerSequence method.....	152
5.5.10.5. GetTriggerSource method.....	153
5.5.10.6. SetTriggerSource method.....	154
5.5.10.7. GetTriggerAdditionalParameterMinMax method.....	155
5.5.10.8. GetTriggerAdditionalParameter method.....	156
5.5.10.9. SetTriggerAdditionalParameter method.....	157
5.5.10.10. GetTriggerDelayMinMax method.....	158
5.5.10.11. GetTriggerDelay method.....	159
5.5.10.12. SetTriggerDelay method.....	160
5.5.10.13. ExecuteSoftwareTrigger method.....	161
5.5.11. ExposureTime methods.....	162
5.5.11.1. GetExposureTimeControl method.....	162
5.5.11.2. SetExposureTimeControl method.....	168
5.5.11.3. GetExposureTimeMinMax method.....	174
5.5.11.4. GetExposureTime method.....	175
5.5.11.5. SetExposureTime method.....	176
5.5.12. DigitalIoControl methods.....	178
5.5.12.1. GetLineModeAll method.....	178
5.5.12.2. SetLineModeAll method.....	179
5.5.12.3. GetLineInverterAll method.....	180
5.5.12.4. SetLineInverterAll method.....	181
5.5.12.5. GetLineStatusAll method.....	182
5.5.12.6. GetUserOutputValueAll method.....	183
5.5.12.7. SetUserOutputValueAll method.....	184
5.5.12.8. GetLineSource method.....	185
5.5.12.9. SetLineSource method.....	189
5.5.13. TimerControl methods.....	193
5.5.13.1. GetTimerDurationMinMax method.....	193
5.5.13.2. GetTimerDuration method.....	194
5.5.13.3. SetTimerDuration method.....	195
5.5.13.4. GetTimerDelayMinMax method.....	196
5.5.13.5. GetTimerDelay method.....	197
5.5.13.6. SetTimerDelay method.....	198
5.5.13.7. GetTimerTriggerSource method.....	199
5.5.13.8. SetTimerTriggerSource method.....	200
5.5.14. Gain methods.....	201
5.5.14.1. GetGainMinMax method.....	201
5.5.14.2. GetGain method.....	202
5.5.14.3. SetGain method.....	203
5.5.14.4. GetGainAuto method.....	204
5.5.14.5. SetGainAuto メソッド.....	205
5.5.15. BlackLevel methods.....	206
5.5.15.1. GetBlackLevelMinMax method.....	206
5.5.15.2. GetBlackLevel method.....	207
5.5.15.3. SetBlackLevel method.....	208

---

---

5.5.16. Gamma methods .....	209
5.5.16.1. GetGammaMinMax method.....	209
5.5.16.2. GetGamma method .....	210
5.5.16.3. SetGamma method.....	211
5.5.17. WhiteBalance methods .....	212
5.5.17.1. GetBalanceRatioMinMax method .....	212
5.5.17.2. GetBalanceRatio method.....	213
5.5.17.3. SetBalanceRatio method .....	214
5.5.17.4. GetBalanceWhiteAuto method .....	215
5.5.17.5. SetBalanceWhiteAuto method.....	216
5.5.18. Hue methods .....	217
5.5.18.1. GetHueMinMax method.....	217
5.5.18.2. GetHue method .....	218
5.5.18.3. SetHue method .....	219
5.5.19. Saturation methods .....	220
5.5.19.1. GetSaturationMinMax method.....	220
5.5.19.2. GetSaturation method.....	221
5.5.19.3. SetSaturation method .....	222
5.5.20. Sharpness methods .....	223
5.5.20.1. GetSharpnessMinMax method .....	223
5.5.20.2. GetSharpness method.....	224
5.5.20.3. SetSharpness method .....	225
5.5.21. ColorCorrectionMatrix methods.....	226
5.5.21.1. GetColorCorrectionMatrixMinMax method .....	226
5.5.21.2. GetColorCorrectionMatrix method .....	228
5.5.21.3. SetColorCorrectionMatrix method .....	229
5.5.22. LUT Control methods .....	230
5.5.22.1. GetLutEnable method.....	230
5.5.22.2. SetLutEnable method .....	231
5.5.22.3. GetLutValue method .....	232
5.5.22.4. SetLutValue method .....	233
5.5.23. UserSetControl methods .....	234
5.5.23.1. ExecuteUserSetLoad method.....	234
5.5.23.2. ExecuteUserSetSave method .....	235
5.5.23.3. ExecuteUserSetQuickSave method .....	236
5.5.23.4. GetUserSetDefault method .....	237
5.5.23.5. SetUserSetDefault method .....	238
5.5.23.6. ExecuteUserSetSaveAndSetDefault method .....	239
5.5.24. SequentialShutterControl methods .....	240
5.5.24.1. GetSequentialShutterEnable method .....	240
5.5.24.2. SetSequentialShutterEnable method .....	241
5.5.24.3. GetSequentialShutterTerminateAtMinMax method .....	242
5.5.24.4. GetSequentialShutterTerminateAt method .....	243
5.5.24.5. SetSequentialShutterTerminateAt method .....	244
5.5.24.6. GetSequentialShutterIndexMinMax method.....	245
5.5.24.7. GetSequentialShutterEntryMinMax method .....	246
5.5.24.8. GetSequentialShutterEntry method.....	247
5.5.24.9. SetSequentialShutterEntry method .....	248
5.5.25. UserDefinedName (DeviceUserID) method .....	249

---

5.5.25.1. GetUserDefinedName method .....	249
5.5.25.2. SetUserDefinedName method.....	250
5.5.26. Chunk methods .....	251
5.5.26.1. GetChunkModeActive method.....	251
5.5.26.2. SetChunkModeActive method .....	252
5.5.27. Properties .....	253
5.5.27.1. Parent property .....	253
5.6. GenApiWrapper class .....	254
5.6.1. INode methods .....	256
5.6.1.1. GetNode method .....	256
5.6.1.2. GetNodeType method.....	257
5.6.1.3. GetName method .....	258
5.6.1.4. GetAccessMode method .....	259
5.6.1.5. GetVisibility method .....	260
5.6.1.6. GetCachingMode method.....	261
5.6.1.7. GetDescription method.....	262
5.6.1.8. GetToolTip method.....	263
5.6.1.9. GetRepresentation method.....	264
5.6.1.10. GetUnit method.....	265
5.6.2. ICategory node .....	266
5.6.2.1. GetNumOfFeatures method .....	266
5.6.2.2. GetFeaturesByIndex method.....	267
5.6.3. IInteger node .....	268
5.6.3.1. GetIntMin method .....	268
5.6.3.2. GetIntMax method .....	269
5.6.3.3. GetIntInc method .....	270
5.6.3.4. GetIntValue method .....	271
5.6.3.5. SetIntValue method .....	272
5.6.4. IFloat node.....	273
5.6.4.1. GetFloatMin method .....	273
5.6.4.2. GetFloatMax method .....	274
5.6.4.3. GetFloatHasInc method.....	275
5.6.4.4. GetFloatInc method .....	276
5.6.4.5. GetFloatDisplayNotation method.....	277
5.6.4.6. GetFloatDisplayPrecision method .....	278
5.6.4.7. GetFloatValue method .....	279
5.6.4.8. SetFloatValuemethod .....	280
5.6.5. IBoolean node .....	281
5.6.5.1. GetBoolValue method.....	281
5.6.5.2. SetBoolValue method .....	282
5.6.6. IEnumeration node .....	283
5.6.6.1. GetNumOfEnumEntries method.....	283
5.6.6.2. GetEnumIntValue method.....	284
5.6.6.3. SetEnumIntValue method .....	285
5.6.6.4. GetEnumStrValue method .....	286
5.6.6.5. SetEnumStrValue method.....	287
5.6.6.6. GetEnumEntryByIndex method .....	288
5.6.7. IEnumEntry node.....	289
5.6.7.1. GetEnumEntryIntValue method .....	289

5.6.7.2. GetEnumEntryStrValue method.....	290
5.6.8. ICommand node .....	291
5.6.8.1. ExecuteCommand method .....	291
5.6.8.2. IsCommandDone method.....	292
5.6.9. IString node .....	293
5.6.9.1. GetStrValue method.....	293
5.6.9.2. SetStrValue method .....	294
5.6.10. Properties .....	295
5.6.10.1. Parent property .....	295
5.7. CamSystemInfo class .....	296
5.7.1. Member Variables (Fields) .....	297
5.7.1.1. gevInfo field .....	297
5.7.1.2. u3vInfo field .....	297
5.7.2. Properties .....	298
5.7.2.1. APIDllVersion property.....	298
5.7.2.2. APIDNetDllVersion property .....	298
5.8. GevSystemInfo class .....	299
5.8.1. Properties .....	300
5.8.1.1. DllExVersion property .....	300
5.8.1.2. DllVersion property .....	300
5.8.1.3. DriverVersion property .....	301
5.9. U3vSystemInfo class.....	302
5.9.1. Properties .....	303
5.9.1.1. DllExVersion property .....	303
5.9.1.2. DllVersion property .....	303
5.9.1.3. DriverVersion property .....	304
5.10. CameralInfo class .....	305
5.10.1. Member Variables (Fields) .....	306
5.10.1.1. gevCamInfo field .....	306
5.10.1.2. u3vCamInfo field .....	306
5.10.2. Properties .....	307
5.10.2.1. CamType property .....	307
5.10.2.2. Manufacturer property .....	307
5.10.2.3. ModelName property .....	308
5.10.2.4. SerialNumber property.....	308
5.10.2.5. UserDefinedName property .....	309
5.11. GevCameralInfo class.....	310
5.11.1. Properties .....	311
5.11.1.1. AdapterDisplayName property .....	311
5.11.1.2. AdapterGateway property.....	311
5.11.1.3. AdapterIPAddress property.....	312
5.11.1.4. AdapterMACAddress property.....	312
5.11.1.5. AdapterSubnet property.....	313
5.11.1.6. CurrentIP_DHCP property .....	313
5.11.1.7. CurrentIP_LLA property .....	314
5.11.1.8. CurrentIP_Persistent property .....	314
5.11.1.9. DisplayName property .....	315
5.11.1.10. Gateway property .....	315
5.11.1.11. IPAddressproperty .....	316



---

5.11.1.12. MACAddress property .....	316
5.11.1.13. Subnet property .....	317
5.11.1.14. SupportIP_DHCP property .....	317
5.11.1.15. SupportIP_LLA property .....	318
5.11.1.16. SupportIP_Persistent property .....	318
5.12. U3vCameraInfo class .....	319
5.12.1. Properties .....	320
5.12.1.1. AdapterDeviceID property .....	320
5.12.1.2. AdapterDfltMaxPacketSize property .....	320
5.12.1.3. AdapterVendorID property .....	321
5.12.1.4. DeviceVersion property .....	321
5.12.1.5. FamilyName property .....	322
5.12.1.6. ManufacturerInfo property .....	322
5.13. CameraImageInfo class .....	323
5.13.1. Properties .....	324
5.13.1.1. Timestamp property .....	324
5.13.1.2. PixelFormat property .....	324
5.13.1.3. SizeX property .....	325
5.13.1.4. SizeY property .....	325
5.13.1.5. OffsetX property .....	326
5.13.1.6. OffsetY property .....	326
5.13.1.7. PaddingX property .....	327
5.13.1.8. BlockID property .....	327
5.13.1.9. BufferPointer property .....	328
5.13.1.10. Size property .....	328
5.13.1.11. ImageID property .....	329
5.13.1.12. Status property .....	329
5.14. GenApiNode class .....	330
5.15. ImageAcquiredEventArgs class .....	331
5.15.1. Properties .....	332
5.15.1.1. BufferIndex property .....	332
5.15.1.2. ImageInfo property .....	332
5.16. ImageErrorReceivedEventArgs class .....	333
5.16.1. Properties .....	334
5.16.1.1. BufferIndex property .....	334
5.16.1.2. ErrorStatus property .....	334
5.17. BufferBusyReceivedEventArgs class .....	335
5.17.1. Properties .....	336
5.17.1.1. BufferIndex property .....	336
5.18. CameraUtility class .....	337
5.18.1. Image format converter .....	338
5.18.1.1. PrepareLUT method .....	338
5.18.1.2. Convert*ToBGR methods .....	339
5.18.1.3. Convert*ToBGR method .....	341
5.18.1.4. ConvertImage .....	343
5.18.2. Miscellaneous utilities .....	345
5.18.2.1. GetBitPerPixel method .....	345
5.18.2.2. GetDataDepth method .....	346
5.18.2.3. IsMonochromic method .....	347

---

---

5.18.2.4. IsPixelBayer methos .....	348
5.18.2.5. SaveBmp* method .....	349
5.18.2.6. ReverselImage method .....	351
5.18.2.7. CopyMemoryT method .....	353
6. EventHandler delegates .....	354
6.1.1. RemovedEventHandler delegate .....	355
6.1.2. ImageAcquiredEventHandler delegate .....	356
6.1.3. ImageErrorReceivedEventHandler delegate .....	357
6.1.4. BufferBusyReceivedEventHandler delegate .....	358
6.1.5. CameraEventReceivedEventHandler delegate .....	359
7. Enumeration types .....	360
7.1. CamApiStatus .....	361
7.2. CameraType .....	362
7.3. CameraAccessmode .....	363
7.4. CameraAcquisitionMode .....	363
7.5. CameraEventType .....	364
7.6. NodeType .....	364
7.7. NodeAccessMode .....	365
7.8. NodeVisibility .....	365
7.9. NodeCachingMode .....	365
7.10. NodeRepresentation .....	366
7.11. NodeFloatNotationn .....	366
7.12. CameraImageFormat .....	366
7.13. CameraPixelFormat .....	367
7.14. CameraTestPattern .....	368
7.15. CameraAcqFrameRateCtrl .....	368
7.16. CameraImageBufferMode .....	368
7.17. CameraTriggerSequence .....	369
7.18. CameraTriggerSource .....	369
7.19. CameraExposureTimeCtrl .....	370
7.20. CameraLineSelector .....	370
7.21. CameraLineSource .....	371
7.22. CameraTimerTriggerSource .....	371
7.23. CameraGainAuto .....	371
7.24. CameraBalanceRatioSelector .....	372
7.25. CameraBalanceWhiteAuto .....	372
7.26. CameraColorCorrectionMatrixSelector .....	372
7.27. CameraUserSetSelector .....	373
7.28. DstPixelFormat .....	373
8. Others .....	374
8.1. Revision History .....	374
8.2. Disclaimer .....	374
8.3. License .....	375
8.4. Inquiry .....	375

---

---

# 1. Introduction

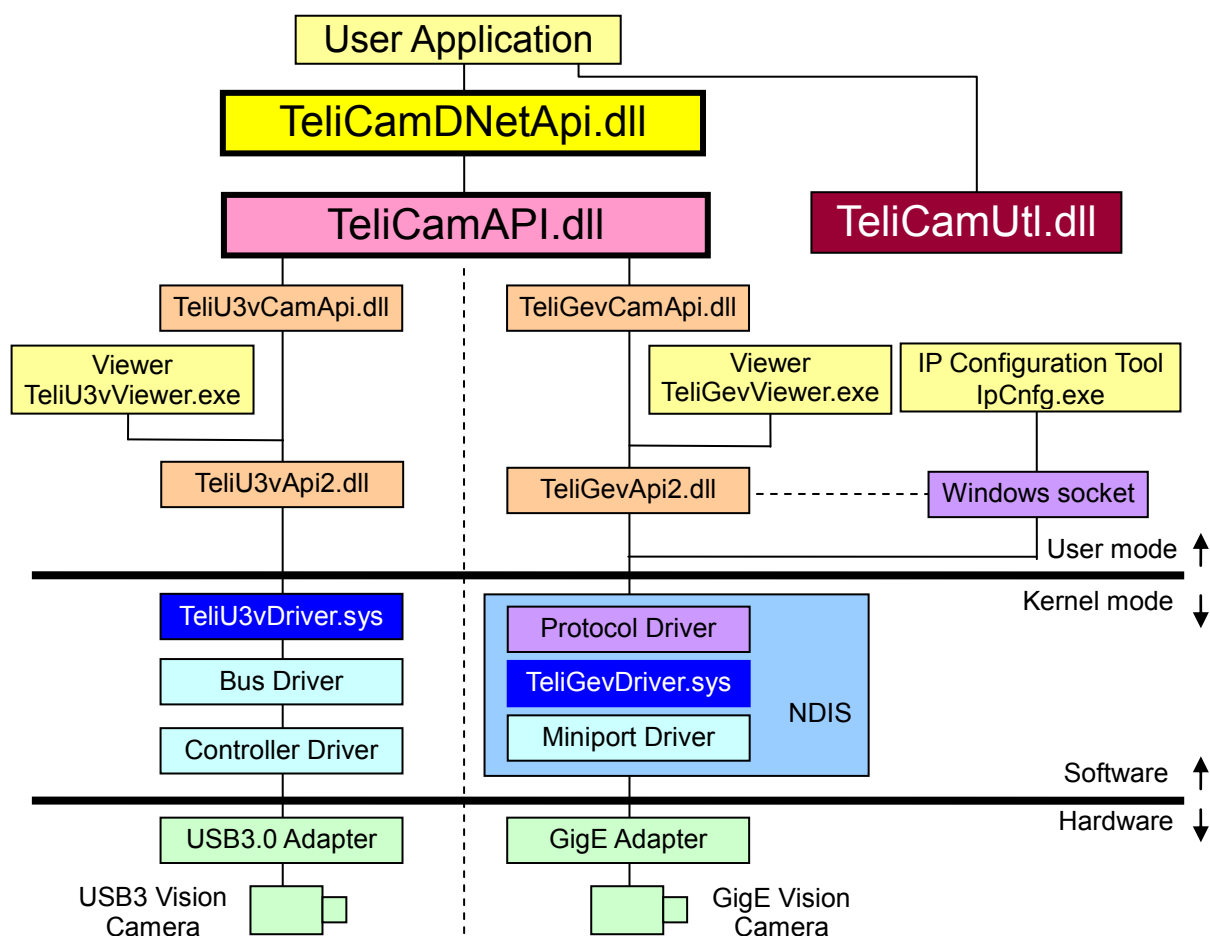
TeliCamSDK is a software development kit to control Toshiba Teli USB3 Vision digital camera series and Gig-E Vision digital camera series from PCs.

This document describes how to use TeliCamDNetAPI, which is a programming interface for Microsoft .NET Framework in the TeliCamSDK package. TeliCamDNetAPI allows users to create .NET Framework applications easily, without paying attention to the camera interface type.

Software engineers who build the systems using cameras are assumed to be readers of this document.

## 2. Configuration

The following figure indicates software configuration of TeliCamDNetSDK.



for 32bit OS	for 64bit OS	Description
TeliCamDNet2_0Api.dll TeliCamDNet4_0Api.dll	TeliCamDNet2_0Api64.dll TeliCamDNet4_0Api64.dll	Class library for designing .NET Framework applications, based on APIs of TeliCamAPI.dll, Refer to 4.5 Developing .NET Framework application.
TeliCamAPI.dll	TeliCamAPI64.dll	Function library for designing native applications.
TeliU3vCamApi.dll	TeliU3vCamApi_64.dll	Extended library for USB3 Vision cameras.
TeliU3vApi2.dll	TeliU3vApi2_64.dll	Function library for USB3 Vision cameras.
TeliU3vDriver.sys	TeliU3vDriver64.sys	Device driver for the USB3 Vision camera.
TeliGevCamApi.dll	TeliGevCamApi64.dll	Extended library for GigE Vision cameras.
TeliGevApi2.dll	TeliGevApi2_64.dll	Function library for GigE Vision cameras.
TeliGevDriver.sys	TeliGevDriver64.sys	Device driver for the GigE Vision camera.
TeliCamUtl.dll	TeliCamUtl64.dll	Utility function library for handling images.
TeliU3vViewer.exe	TeliU3vViewer64.exe	Viewer for checking features and images of the USB3 Vision camera.
TeliGevViewer.exe	TeliGevViewer64.exe	Viewer for checking features and images of the GigE Vision camera.
IpCnfg.exe	IPCnfg.exe	Tool for setting the IP addresses of cameras.

Note that TeliCamDNetApi (TeliCamAPI) does not support multi-cast feature of GigE Vision camera.

---

## 3. Operation Environment

The following table shows environments necessary for using TeliCamDNetAPI.  
This environment, however, does not always guarantee operation of all application programs.  
A higher performance host PC may be required depending on use conditions.

Supported OS	<ul style="list-style-type: none"><li>● Windows 7 32/64-bit</li><li>● Windows 8.1 32/64bit</li><li>● Windows 10 32/64-bit</li></ul>
.NET Framework	<ul style="list-style-type: none"><li>● Microsoft .NET Framework 2.0 SP2 or Microsoft .NET Framework 4.0</li></ul>
Recommended PC Specifications	<ul style="list-style-type: none"><li>● CPU : Intel Core2 2.40GHz or above recommended</li><li>● Memory : 2Gbytes or above</li><li>● Graphics : VRAM with 256 Mbytes or above mounted</li></ul>
Recommended USB3.0 Adapter	<ul style="list-style-type: none"><li>● USB3.0 adapter with USB3.0 host controller from Renesas Electronics *1</li></ul>
Recommended Network Adapter	<ul style="list-style-type: none"><li>● Gigabit Ethernet adapter supporting Jumbo Frame (Jumbo Packet) (9014 bytes or above) (Intel PRO/1000 series, etc.) *2</li></ul>
Required Software	<ul style="list-style-type: none"><li>● Microsoft Visual C++ 2010 SP1 Redistributable Package</li><li>● GenICam GenApi reference implementation v.3.0.1 *3,*4</li><li>● Microsoft Direct X End-User Runtime (DirectX 9.0c or later) *4</li></ul>
Supported Camera	<ul style="list-style-type: none"><li>● Toshiba Teli USB3 Vision digital camera</li><li>● Toshiba Teli Gig-E Vision digital camera</li></ul>

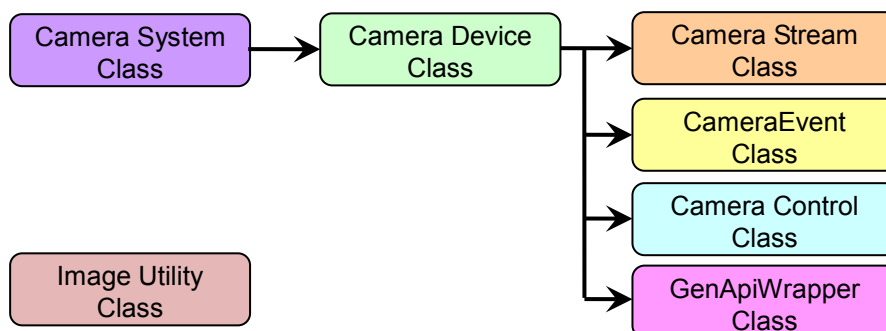
### Notes

- \*1: Necessary in application that uses USB3 Vision cameras.
- \*2: Necessary in application that uses GigE Vision cameras.
- \*3: Necessary in development environment even if GenApi module is disabled in user application.
- \*4: Necessary in TeliU3vViewer and TeliGevViewer.

## 4. Library

TeliCamDNetAPI is a class library that wraps TeliCamAPI function library. Refer to section “4.1 Usage” in “[TeliCamAPI Library Manual Eng.pdf](#)” about basic usage of APIs.

The following diagram shows class hierarchy of TeliCamDNetAPI library.



Class	Description
CameraSystem class	Class for controlling TeliCamDNetAPI system itself.
CameraDevice class	Class for controlling a camera. This class contains Camera Stream class, Camera Event class, Camera Control class, and GenApi Wrapper class objects as its members.
CameraStream class	Class for controlling an image stream interface.
CameraEvent class	Class for notifying events of the camera.
CameraControl class	Class for controlling individual features of the camera.
GenApiWrapper class	Class for controlling individual features of the camera using GenApi module. User application can control features or get information that Control class does not support.
CameraUtility class	Class for providing utility functions for handling images.

Note:

- ✓ Current version TeliCamDNetAPI does not provide wrapper class of Low-level stream APIs and Low-level event API, neither.
- ✓ Some CameraControl class methods use direct camera register access instead of GenApi module. CameraControl class methods may realize the higher performance than GenApiWrapper class methods

### 4.1. Wrapping TeliCamAPI functions

The following table shows TeliCamAPI functions that TeliCamDNetAPI methods wraps.

TeliCamDNetAPI class	TeliCamDNetAPI method	TeliCamAPI function
CameraSystem	Initialize	Sys_Initialize
	Terminate	Sys_Terminate
	GetInformation	Sys_GetInformation
	GetNumOfCameras	Sys_GetNumOfCameras
	GetCameraInformation	Cam_GetInformation
	CreateDeviceObject	

TeliCamDNetAPI class	TeliCamDNetAPI method	TeliCamAPI function
	CreateDeviceObjectFromInfo	
CameraDevice	GetInformation	Cam_GetInformation
	Open	Cam_Open
	Close	Cam_Close
	ReadRegister	Cam_ReadReg
	WriteRegister	Cam_WriteReg
	ResetPort	Cam_ResetPort
	GetHeartbeat	Cam_GetHeartbeat
	SetHeartbeat	Cam_SetHeartbeat
	GetTLParamsLocked	GetCamTLParamsLocked
	GetLastGenlCamError	Misc_GetLastGenlCamError
CameraStream	Open	Strm_OpenSimple
	Close	Strm_Close
	Start	Strm_Start
	Stop	Strm_Stop
	ReadCurrentImage	Strm_ReadCurrentImage
	GetCurrentBufferIndex	Strm_GetCurrentBufferIndex
	LockBuffer	Strm_LockBuffer
	UnlockBuffer	Strm_UnlockBuffer
CameraEvent	Open	Evt_OpenSimple
	Close	Evt_Close
	Activate	Evt_Activate
	Deactivate	Evt_Deactivate
CameraControl	Set***	SetCam***
	Get***	GetCam***
	Execute***	ExecuteCam***
GenApiWrapper	GetNodeType	Nd_GetType
	Get***	Nd_Get***
	Set***	Nd_Set***
	ExecuteCommand	Nd_CmdExecute
	IsCommandDone	Nd_GetCmdIsDone
CameraUtility	PrepareLUT	PrepareLUT
	GetBitPerPixel	BitPerPixel
	GetDataDepth	DataDepth
	IsMonochromic	IsMonochromic
	IsPixelBayer	IsPixelBayer
	SaveBmpARGB	SaveBmpARGB
	SaveBmpRGB	SaveBmpRGB
	SaveBmpMono	SaveBmpMono
	ReverseImage	ReverseImg
	Convert***	Conv***

---

## 4.2. Difference of usage between TeliCamAPI functions

The usage of TeliCamDNetAPI, the wrapper class library of TeliCamAPI, is almost same as the usage of TeliCamAPI functions.

Refer to "[TeliCamAPI Library Manual Eng.pdf](#)" about basic usage of APIs.

This section describes about difference of usage between TeliCamDNetAPI and TeliCamAPI.

Item	TeliCamAPI	TeliCamDNetAPI
Opening a camera. (with camera index)	Call " <u>Cam_Open()</u> " function with specifying camera index as a parameter.	Call " <u>Open()</u> " method of Camera Device class instance, <u>after creating the CameraDevice instance using "CreateDeviceObject()" method of CameraSystem class with specifying specified camera index as a parameter.</u>
Opening a camera. (With serial number, model name, etc.)	Call " <u>Cam_OpenFromInfo()</u> " function with specifying serial number or other information as a parameter.	Call " <u>Open()</u> " method of Camera Device class instance, <u>after creating the CameraDevice instance using "CreateDeviceObjectFromInfo()" method of CameraSystem class with specifying serial number or other information as a parameter.</u>
Acquiring stream (image) data.	Call " <u>Strm_SetCallbackImageAcquired()</u> " function to register a <u>callback function</u> , which will be called on reception of an image.	<u>Register delegate for a method</u> , that processes received image, <u>to the event handler of "ImageAcquired" event</u> (CameraStream class).  "ImageAcquired" event will be raised on reception of an image and the registered method will be executed.
	<u>Event object</u> is also available for detecting reception of an image.  An event object will be set signaled on reception of an image, if the event object is registered to TeliCamAPI as "hCmpEvt" argument on calling " <u>Strm_OpenSimple()</u> " function.	<u>AutoResetEvent object</u> is available for detecting reception of an image.  An AutoResetEvent object will be set signaled on reception of an image, if the object is registered to CameraStream object as "receivedEvent" argument on calling " <u>Open()</u> " method of CameraStream class.
	Low-level functions are <u>available</u> .	Methods wrapping Low-level functions are <u>not provided</u> .



Item	TeliCamAPI	TeliCamDNetAPI
Acquiring CameraEvent (message) data.		<p><u>CameraEvent</u> class provides event for each camera-event.</p> <p>Register delegate for a method, that processes received event data, to event handler of an event in <u>CameraEvent</u> object and call "<u>Activate()</u>" method with name of the event to activate the event.</p>
	<p><u>Event object</u> is also available for detecting reception of a camera-event.</p> <p>An <u>event object</u> will be set signaled on reception of a camera-event (message), if the camera-event is activated and the <u>event object</u> is registered to TeliCamAPI by specifying the <u>event object</u> and name of the camera-event (message) as arguments on calling "<u>Evt Activate()</u>" function.</p>	<p><u>AutoResetEvent object</u> is available for detecting reception of a camera-event.</p> <p>An <u>AutoResetEvent object</u> will be set signaled on reception of a camera-event, if the camera-event is activated and the <u>AutoResetEvent object</u> is registered to CameraEvent object by specifying the <u>AutoResetEvent object</u> and name of the camera-event (message) as arguments on calling "<u>Activate()</u>" method of CameraEvent object.</p>
	Low-level functions are <u>available</u> .	Methods wrapping Low-level functions are <u>not provided</u> .

---

## 4.3. Installing SDK

Please refer to PDF document "[Start-up Guide Eng](#)" in "Documents" folder under the TeliCamSDK folder to install TeliCamSDK.

## 4.4. Uninstalling SDK

Please refer to PDF document "[Start-up Guide Eng](#)" in "Documents" folder under the TeliCamSDK folder to uninstall TeliCamSDK.

## 4.5. Developing .NET Framework application

C#, VB.NET, C++/CLI languages are available for developing .NET Framework applications.

TeliCamSDK provides 4 TeliCamDNetAPI dlls. Use one of four dlls in your projects.

<div><div>.NET Framework</div><div>CPU type</div></div>	.NET Framework 2.0, 3.0, 3.5	.NET Framework 4.0, 4.5, 4.6, 4.7
x86	[Install Folder] / bin / <b>x86</b> / TeliCamDNet <b>2_0</b> Api.dll	[Install Folder] / bin / <b>x86</b> / TeliCamDNet <b>4_0</b> Api.dll
x64	[Install Folder] / bin / <b>x64</b> / TeliCamDNet <b>2_0</b> Api <b>64</b> .dll	[Install Folder] / bin / <b>x64</b> / TeliCamDNet <b>4_0</b> Api <b>64</b> .dll


















---

## 5. Classes in TeliCamDNetAPI


The following table shows classes that TeliCamDNetAPI provides

### [Classes]

namespace: Teli.TeliCamAPI.NET.

	Name	Description
	CameraSystem	The root system class of TeliCamDNetAPI.
	CameraDevice	Class for controlling a camera.
	CameraStream	Class for controlling a (image) stream.
	CameraEvent	Class for controlling camera-events of a camera.
	CameraControl	Class for controlling features of a camera.
	GenApiWrapper	Class for controlling features of a camera using GenICam GenApi module.
	CamSystemInfo	Class for information of TeliCamDNetAPI system.
	GevSystemInfo	Class for information of GigE Vision API system.
	U3vSystemInfo	Class for information of USB3 Vision API system.
	CameraInfo	Class for information of a camera.
	GevCameraInfo	Class for information of a GigE Vision camera.
	U3vCameraInfo	Class for information of a USB3 Vision camera.
	CameraImageInfo	Class for information of an image received from the camera.
	GenApiNode	Node class of camera description data, used in GenApi Wrapper class.
	CameraStream. ImageAcquiredEventArgs	Event argument class for ImageAcquired event.
	CameraStream. ImageErrorReceivedEventArgs	Event argument class for ImageErrorReceived event.
	CameraStream. BufferBusyReceivedEventArgs	Event argument class for BufferBusyReceived event.

namespace: Teli.TeliCamAPI.NET.Utility

	Name	Description
	CameraUtility	Class for providing image handling utility methods.

### [Remarks]

Classes, methods, properties not described in this manual are for use inside TeliCamDNetAPI only.

---

## 5.1. CameraSystem class

This class is the root system class of TeliCamDNetAPI.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

---

[C#]

```
public class CameraSystem : IDisposable
```

[VB.NET]

```
Public Class CameraSystem  
    Implements IDisposable
```


[C++]

```
public ref class CameraSystem : IDisposable
```

The followings are members of CameraSystem class available for user applications.










### [Constructors]

---

	Name	Description
	CameraSystem	Initializes a new instance of the CameraSystem class.

### [Methods]

---

	Name	Description
	Dispose	Releases all resources used by the CameraSystem.
	Initialize()	Initializes CameraSystem class instance targeting all camera interface types.
	Initialize(CameraType)	Initializes CameraSystem class instance specifying camera interface type(s).
	Terminate	Terminates CameraSystem class instance.
	GetInformation	Retrieves system information of TeliCamDNetSDK.
	GetNumOfCameras	Retrieves number of detected cameras.
	GetCameraInformation	Retrieves information of a detected camera.
	CreateDeviceObject	Retrieves CameraDevice class instance of a camera, whose camera index in the detected camera list is the specified value.
	CreateDeviceObjectFromInfo	Retrieves CameraDevice class instance of a camera that has specified camera information value.

---

### 5.1.1. Constructors

Initializes CameraSystem class instance.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public CameraSystem ()
```

[VB.NET]

```
Public Sub New
```

[C++]

```
public:  
CameraSystem ()
```

---

## 5.1.2. Methods

### 5.1.2.1. Dispose method

This method releases all resources used by the CameraSystem.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public void Dispose()
```

[VB.NET]

```
Public Sub Dispose
```

[C++]

```
public:
```

```
virtual void Dispose() sealed
```

---

### 5.1.2.2. Initialize method

This method initializes CameraSystem class instance targeting all camera interface types.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus Initialize()  
  
public CamApiStatus Initialize(  
    CameraType cameraType  
)
```

[VB.NET]

```
Public Function Initialize As CamApiStatus  
  
Public Function Initialize (  
    cameraType As CameraType  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Initialize()  
  
public:  
CamApiStatus Initialize(  
    CameraType cameraType  
)
```

#### [Parameters]

---

Parameters	Description
<i>cameraType</i>	Interface type of target cameras. CameraType.TypeAll is specified in case of Initialize method with no arguments.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

User application must call this function once, before calling any other methods in TeliCamDNetApi.

“|” (OR) is available for specifying plural types of camera to [cameraType](#) parameter.

The following is example.

```
CameraSystemObject.Initialize(CameraType.TypeGev | CameraType.TypeU3v);
```

Specifying [cameraType](#) parameter in this method may make processing time of methods shorter.

---

### 5.1.2.3. Terminate method

This method terminates CameraSystem class instance.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public CamApiStatus Terminate()
```

[VB.NET]

```
Public Function Terminate As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Terminate()
```

#### **[Return value]**

---

Returns result status.

#### **[Remarks]**

---

User application must call this function once, before disposing CameraSystem object.  
This method will be executed in the destructor of CameraSystem if it has not been called.



---

#### 5.1.2.4. GetInformation method

This function retrieves system information of TeliCamDNetSDK.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public CamApiStatus GetInformation(  
    ref CamSystemInfo sysInfo  
)
```

[VB.NET]

```
Public Function GetInformation (  
    ByRef sysInfo As CamSystemInfo  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetInformation(  
        CamSystemInfo^% sysInfo  
)
```

##### [Parameters]

---

Parameters	Description
<i>sysInfo</i>	A variable that receives the acquired system information.

##### [Return value]

---

Returns result status.

---

### 5.1.2.5. GetNumOfCameras method

This method creates list of connected cameras in TeliCamDNetAPI (TeliCamAPI) system and returns number of detected cameras.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetNumOfCameras(  
    out int num  
)
```

[VB.NET]

```
Public Function GetNumOfCameras (  
    <OutAttribute> ByRef num As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetNumOfCameras(  
    [OutAttribute] Int32% num  
)
```

#### [Parameters]

---

Parameters	Description
<i>num</i>	A variable that receives the number of detected cameras.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

Camera index, whose value is up to ([num](#) – 1) from 0, is assigned to each detected camera for identifying it.

User application cannot get [CameraDevice](#) object using “[CreateDeviceObject\(\)](#)” method or “[CreateDeviceObjectFromInfo\(\)](#)” method until this function is called, because camera list inside TeliCamDNetAPI (TeliCamAPI) is not ready.

---

### 5.1.2.6. GetCameraInformation method

This method retrieves information of a detected camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetCameraInformation(  
    int cameraIndex,  
    ref CameraInfo camInfo  
)
```

[VB.NET]

```
Public Function GetCameraInformation (  
    cameraIndex As Integer,  
    ByRef camInfo As CameraInfo  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetCameraInformation(  
    Int32 cameraIndex,  
    CameraInfo^% camInfo  
)
```

#### [Parameters]

---

Parameters	Description
<i>cameraIndex</i>	Index of camera. Index value should be up to number of detected cameras minus 1, from 0.
<i>camInfo</i>	A variable that receives retrieved camera information.

#### [Return value]

---

Returns result status.

---

### 5.1.2.7. CreateDeviceObject method

This method retrieves [CameraDevice](#) class instance of a camera, whose camera index in the detected camera list is the specified value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus CreateDeviceObject(  
    int cameraIndex,  
    ref CameraDevice camera  
)
```

[VB.NET]

```
Public Function CreateDeviceObject (  
    cameraIndex As Integer,  
    ByRef camera As CameraDevice  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus CreateDeviceObject(  
    Int32 cameraIndex,  
    CameraDevice^% camera  
)
```

#### [Parameters]

---

Parameters	Description
<i>cameraIndex</i>	Index of camera. Index value should be up to number of detected cameras minus 1, from 0.
<i>Camera</i>	A variable that receives created CameraDevice object.

#### [Return value]

---

Returns result status.

---

### 5.1.2.8. CreateDeviceObjectFromInfo method

This method retrieves [CameraDevice](#) class instance of a camera that has specified camera information value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus CreateDeviceObjectFromInfo(  
    string serialNo,  
    string modelName,  
    string userDefinedName,  
    ref CameraDevice camera  
)
```

[VB.NET]

```
Public Function CreateDeviceObjectFromInfo (  
    serialNo As String,  
    modelName As String,  
    userDefinedName As String,  
    ByRef camera As CameraDevice  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus CreateDeviceObjectFromInfo(  
    String^ serialNo,  
    String^ modelName,  
    String^ userDefinedName,  
    CameraDevice^% camera  
)
```

#### [Parameters]

---

Parameters	Description
<i>serialNo</i>	Serial number of the camera.
<i>modelName</i>	Model name of the camera.
<i>userDefinedName</i>	User defined name of the camera.
<i>camera</i>	A variable that receives created CameraDevice object.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

Specify null (C#: null, VB.NET: Nothing, C++: nullptr) to [serialNo](#), [modelName](#), [userDefinedName](#) if the parameter is not used for specifying a camera.

If multiple cameras have specified parameters, this method returns a camera detected earliest.

---

## 5.2. CameraDevice class

CameraDevice is a class for controlling a camera.

Use “[CreateDeviceObject\(\)](#)” or “[CreateDeviceObjectFromInfo\(\)](#)” method of [CameraSystem](#) class to obtain CameraDevice class instance.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

---

[C#]

```
public class CameraDevice : IDisposable
```

[VB.NET]

```
Public Class CameraDevice  
    Implements IDisposable
```

















[C++]

```
public ref class CameraDevice : IDisposable
```

The followings are members of CameraDevice class available for user applications.





### [Methods]

---




	Name	Description
	Dispose	Releases all resources used by the CameraDevice.
	GetInformation	Retrieves information of the camera.
	Open()	Opens the camera.
	Open(Boolean, Byte[], CameraAccessMode)	Opens the camera with parameters.
	Close	Closes the camera.
	ReadRegister(Int64, Int32)	Reads an Int32 type data from camera register.
	ReadRegister(Int64, UInt32, Byte[])	Reads Byte array data from camera register.
	ReadRegister(Int64, UInt32, Int32[])	Reads Int32 array data from camera register.
	WriteRegister(Int64, Int32)	Writes an Int32 type data to camera register.
	WriteRegister(Int64, UInt32, Byte[])	Writes Byte array data to camera register.
	WriteRegister(Int64, UInt32, Int32[])	Writes Int32 array data to camera register.
	ResetPort	Resets port of host adapter / host controller that the camera is connected.
	GetHeartbeat	Retrieves current Heartbeat state (Enabled or not) of the camera.
	SetHeartbeat	Writes new Heartbeat parameter (Enabled or not) to the camera.
	GetTLParamsLocked	Reads “TLParamsLocked” value.
	GetLastGenICamError	Retrieves information of GenICamError.

---


**[Member Variables (Fields)]**

	Name	Description
	camStream	<a href="#">CameraStream</a> object for receiving images.
	camEvent	<a href="#">CameraEvent</a> object for receiving camera-events.
	camControl	<a href="#">CameraControl</a> object for controlling camera features.
	genApi	<a href="#">GenApiWrapper</a> object for controlling camera features.

**[Properties]**

	Name	Description
	camType	Gets camera interface type.
	camIndex	Gets index of the camera at the timing when <a href="#">CameraDevice</a> object was created.
	IsOpen	Gets a value indicating whether the camera is opened.

**[Events]**

	Name	Description
	Removed	Occurs when the camera is disconnected.

---

## 5.2.1. Methods

### 5.2.1.1. Dispose method

This method releases all resources used by the CameraDevice.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public void Dispose()
```

[VB.NET]

```
Public Sub Dispose
```

[C++]

```
public:
```

```
virtual void Dispose() sealed
```



---

### 5.2.1.2. GetInformation method

This method retrieves information of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetInformation(  
    ref CameraInfo info  
)
```

[VB.NET]

```
Public Function GetInformation (  
    ByRef info As CameraInfo  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetInformation (  
    CameraInfo^% info  
)
```

#### [Parameters]

Parameters	Description
<i>info</i>	A variable that receives information of the camera

#### [Return value]

Returns result status.

---

### 5.2.1.3. Open method

This method opens the camera.

User application can open cameras that the other applications are using.

User application cannot open CameraStream or CameraEvent if the other application is using it.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus Open()  
  
public CamApiStatus Open(  
    bool useGenICam,  
    byte[] xmlData,  
    CameraAccessmode accessMode  
)
```

[VB.NET]

```
Public Function Open As CamApiStatus  
  
Public Function Open (  
    useGenICam As Boolean,  
    xmlData As Byte(),  
    accessMode As CameraAccessmode  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Open()  
  
public:  
CamApiStatus Open(  
    Boolean useGenICam,  
    array<Byte>^ xmlData,  
    CameraAccessmode accessMode  
)
```

#### [Parameters]

---

Parameters	Description
<i>useGenICam</i>	Flag indicating whether activates GenICam methods. GenICam methods will be activated in case of "Open()" method with no parameters.
<i>xmlData</i>	Camera description data (XML data) saved in PC. Specify null (C#: null, VB.NET: Nothing, C++: nullptr) to load xmlData in the camera. Camera description file in the camera will be used in case of "Open()" method with no parameters.

---

<i>accessMode</i>	Access mode (Control Channel Privilege) of the camera. This parameter is used only for GigE Vision camera. <a href="#">CameraAccessmode</a> . Control will be used in case of "Open()" method with no parameters.
-------------------	---

#### **[Return value]**

Returns result status.

#### **[Remarks]**

In USB3 Vision camera case, this method clears "StreamEnable" node (SIControl register) and "EventEnable" node (EICControl register) to 0 (Disable) to stop image streaming and CameraEvent notification, if no other applications are using the camera.

In GigE Vision camera case, TeliCamDNetAPI (TeliCamAPI) will enable Heartbeat process on opening camera, with Heartbeat timeout value 15sec. Heartbeat process in the camera and TeliCamDNetAPI (TeliCamAPI) will periodically check existence of communication. User application may send dummy command to check communication state, if necessary. When no communication state continued, camera or TeliCamDNetAPI (TeliCamAPI) will judge that Heartbeat timeout error occurred. The camera will cancel the privilege given to the host application, on Heartbeat timeout. User application will make [Removed](#) event object signaled on Heartbeat timeout.

If user application is interrupted for a long time, for example, debugging case and so on, the camera will judge that Heartbeat timeout occurred and will cancel control channel privilege, which will cause the situation that the application cannot access the camera after the application resumed.

By disabling Heartbeat process or setting log timeout value using "[SetHeartbeat](#)()" method, camera will not judge that Heartbeat timeout occurred.

Refer to "4.1.7 Camera Access mode (Control Channel Privilege)" in "[TeliCamAPI Library Manual Eng.pdf](#)" about Control Channel Privilege.

Refer to "4.1.8 Heartbeat process" in "[TeliCamAPI Library Manual Eng.pdf](#)" about Heartbeat process.

When true is specified to [useGenICam](#), TeliCamDNetAPI (TeliCamAPI) will load camera description file data (Xml file) and prepare internal data for enabling GenICam function. If it is the first time that the target camera is used in the user application, it may take a few minutes for loading and preparing data.

When false is specified to [useGenICam](#), processing time of opening camera will be shortened, but methods of [GenApiWrapper](#) and [CameraEvent](#) classes, and a part of methods in [CameraControl](#) class will become unavailable.

If user application called a method that is not available because false is specified to [useGenICam](#), the method will return `CamApiStatus.NotAvailable`.

---

#### 5.2.1.4. Close method

This method closes the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public CamApiStatus Close()
```

[VB.NET]

```
Public Function Close As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Close()
```

##### **[Return value]**

---

Returns result status.

##### **[Remarks]**

---

Calling this method during the other thread is using the camera may cause error in the other thread.  
Call this method after all threads finished using the camera.

---

### 5.2.1.5. ReadRegister method

This method reads data from camera register.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus ReadRegister(  
    long address,  
    out int data  
)  
  
public CamApiStatus ReadRegister(  
    long address,  
    int sizeQuadlet,  
    byte[] byteArray  
)  
  
public CamApiStatus ReadRegister(  
    long address,  
    int sizeQuadlet,  
    int[] intArray  
)
```

[VB.NET]

```
Public Function ReadRegister (  
    address As Long,  
    <OutAttribute> ByRef data As Integer  
) As CamApiStatus  
  
Public Function ReadRegister (  
    address As Long,  
    sizeQuadlet As Integer,  
    byteArray As Byte()  
) As CamApiStatus  
  
Public Function ReadRegister (  
    address As Long,  
    sizeQuadlet As Integer,  
    intArray As Integer()  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus ReadRegister(  
    Int64 address,  
    [OutAttribute] Int32% data  
)
```

---

```

public:
    CamApiStatus ReadRegister(
        Int64 address,
        Int32 sizeQuadlet,
        array<Byte>^ byteArray
    )

```

```

public:
    CamApiStatus ReadRegister(
        Int64 address,
        Int32 sizeQuadlet,
        array<Int32>^ intArray
    )

```

#### **[Parameters]**

---

Parameters	Description
<i>address</i>	Start address of registers to read.
<i>sizeQuadlet</i>	Size of data to read, in Quadlet. Quadlet is 4 bytes in size. Specifying 1 means reading 4 bytes.
<i>data</i>	A variable that receives the read data.
<i>byteArray</i>	A byte type data array that receives the read data.
<i>intArray</i>	An Int32 type data array that receives the read data.

#### **[Return value]**

---

Returns result status.

#### **[Remarks]**

---

“ReadRegister()” method, with byte type data array parameter, is available for reading a string from camera register.

“ReadRegister()” method, with Int32 type data array parameter, is available for reading data from a block of registers.

Registers in the camera are aligned to 4-bytes boundary. The accessing unit of camera register data is Quadlet (4 bytes).

GigE Vision camera handles Big-endian data. TeliCamDNetAPI (TeliCamAPI) handles Big-endian byte order register data received from GigE Vision camera as 32bit integer array data. TeliCamDNetAPI (TeliCamAPI) will change byte order of received data within each 4 bytes (Quadlet) range to convert Big-endian integer data to Little-endian data, and write them to ‘byteArray’ or ‘intArray’ parameter of this method. When received data contains data whose data type is not 4 bytes length data type, user application should change byte order of ‘byteArray’ or ‘intArray’ parameter by itself to make the data valid.

---

### 5.2.1.6. WriteRegister method

This method writes data to camera registers.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

```
[C#]  
public CamApiStatus WriteRegister(  
    long address,  
    int data  
)
```

```
public CamApiStatus WriteRegister(  
    long address,  
    int sizeQuadlet,  
    byte[] byteArray  
)
```

```
public CamApiStatus WriteRegister(  
    long address,  
    int sizeQuadlet,  
    int[] intArray  
)
```

#### [VB.NET]

```
Public Function WriteRegister (  
    address As Long,  
    data As Integer  
) As CamApiStatus
```

```
Public Function WriteRegister (  
    address As Long,  
    sizeQuadlet As Integer,  
    byteArray As Byte()  
) As CamApiStatus
```

```
Public Function WriteRegister (  
    address As Long,  
    sizeQuadlet As Integer,  
    intArray As Integer()  
) As CamApiStatus
```

#### [C++]

```
public:  
    CamApiStatus WriteRegister(  
        Int64 address,  
        Int32 data  
)
```

---

```

public:
    CamApiStatus WriteRegister(
        Int64 address,
        Int32 sizeQuadlet,
        array<Byte>^ byteArray
    )

```

```

public:
    CamApiStatus WriteRegister(
        Int64 address,
        Int32 sizeQuadlet,
        array<Int32>^ intArray
    )

```

#### **[Parameters]**

---

Parameters	Description
<i>address</i>	Start address of registers to write.
<i>sizeQuadlet</i>	Size of data to write, in Quadlet. Quadlet is 4 bytes in size. Specifying 1 means writing 4 bytes.
<i>data</i>	A variable that contains new register data.
<i>byteArray</i>	A byte type data array that contains new register data.
<i>intArray</i>	An Int32 type data array that contains new register data.

#### **[Return value]**

---

Returns result status.

#### **[Remarks]**

---

“WriteRegister()” method, with byte type data array parameter, is available for writing a string to camera register.

“WriteRegister()” method, with Int32 type data array parameter, is available for writing data to a block of registers.

Registers in the camera are aligned to 4-bytes boundary. The accessing unit of camera register data is Quadlet (4 bytes).

GigE Vision camera handles Big-endian data. TeliCamDNetAPI (TeliCamAPI) handles ‘byteArray’ or ‘intArray’ parameter of this method as Little-endian byte order 32bit integer array data.

TeliCanDNetAPI (TeliCamAPI) will change byte order of ‘byteArray’ or ‘intArray’ parameter within each 4 bytes (Quadlet) range to convert Little-endian integer data to Bittle-endian data, and send them to the camera. When ‘byteArray’ or ‘intArray’ contains data whose data type is not 4 bytes length data type, user application should change byte order of the data beforehand to send valid data to the camera.



---

#### 5.2.1.7. ResetPort method

This method resets port of host adapter / host controller on the adapter that the camera is connected.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus ResetPort()
```

[VB.NET]

```
Public Function ResetPort As CamApiStatus
```

[C++]

```
public:  
CamApiStatus ResetPort()
```

##### [Return value]

Returns result status.

##### [Remarks]

This method is available only for USB3 Vision camera.

CameraDevice object should be opened, before calling "[ResetPort\(\)](#)" method.

["Removed"](#) event will be raised on executing "ResetPort()" method.

Plug and play service will search USB controller and cameras and attach device driver again, after this method is executed.

It will take for a while for [CameraSystem](#) object to recognize all cameras connected to the reset port.

Calling "[GetNumOfCameras\(\)](#)" before obtaining a CameraDevice object is recommended, to check that all connected cameras are recognized by confirming number of recognized cameras.

Note that most problems will be recovered by calling this method when problem occurred. But there may be problems which will not be recovered by calling this method.

---

### 5.2.1.8. GetHeartbeat method

This method retrieves current Heartbeat state (enabled or not) of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetHeartbeat(  
    out bool enable,  
    out int hbTimeout  
)
```

[VB.NET]

```
Public Function GetHeartbeat (  
    <OutAttribute> ByRef enable As Boolean,  
    <OutAttribute> ByRef hbTimeout As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetHeartbeat(  
    [OutAttribute] Boolean% enable,  
    [OutAttribute] Int32% hbTimeout  
)
```

#### [Parameters]

Parameters	Description
<i>Enable</i>	A variable that receives current Heartbeat enable state.
<i>hbTimeout</i>	A variable that receives current Heartbeat timeout value of the camera, in milliseconds.

#### [Return value]

Returns result status.

#### [Remarks]

This function is available only for GigE Vision camera.

---

### 5.2.1.9. SetHeartbeat method

This method writes new Heartbeat parameter (enabled or not) to the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetHeartbeat(  
    bool enable,  
    int hbTimeout  
)
```

[VB.NET]

```
Public Function SetHeartbeat (  
    enable As Boolean,  
    hbTimeout As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetHeartbeat(  
    Boolean enable,  
    Int32 hbTimeout  
)
```

#### [Parameters]

Parameters	Description
<i>enable</i>	A variable that contains new Heartbeat enable state.
<i>hbTimeout</i>	A variable that contains new Heartbeat timeout value of the camera, in milliseconds. Minimum available value is 500 milliseconds.

#### [Return value]

Returns result status.

#### [Remarks]

This function is available only for GigE Vision camera.

There are cameras whose Heartbeat function cannot be disabled. For example, GiantDragon series. .

TeliCamDNetAPI (TeliCamAPI) will enable Heartbeat function of camera and set timeout period 15 sec, on opening camera if the camera is GigE Vision type.

It will not be necessary to change Heartbeat parameters, usually. Change Heartbeat parameters using this function when user application may be interrupted more than a few seconds, for example debugging user application.

Never change "GevHeartbeatTimeout" register or "GevGCCPHeartbeatDisable" register using "[WriteRegister\(\)](#)" method or methods in [GenApiWrapper](#) class, because TeliCamDNetApi controls it.

---

### 5.2.1.10. GetTLParamsLocked method

This method reads TLParamsLocked value, which is used for protecting transport layer critical registers from changing value during streaming is active.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetTLParamsLocked(  
    out int value  
)
```

[VB.NET]

```
Public Function GetTLParamsLocked (  
    <OutAttribute> ByRef value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTLParamsLocked(  
    [OutAttribute] Int32% value  
)
```

#### [Parameters]

---

Parameters	Description
<i>value</i>	A variable that receives current TLParamsLocked value.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

TLParamsLocked is used for protecting transport layer critical registers from changing value during streaming is active. When TLParamsLocked value is 1, critical parameters of the camera will be locked and commands for editing them will be rejected.

TLParamsLocked value is controlled in TeliCamDNetApi (TeliCamAPI) internally. User application has to do nothing about controlling TLParamsLocked value.

If user application writes new value to camera registers protected by TLParamsLocked using "[WriteRegister\(\)](#)" method, the method may receive success status. Actually, internal writing action may be skipped or writing action may be reserved until TLParamsLocked is cleared.

This method will return error when GenICam function is disabled on opening CameraDevice.

---

### 5.2.1.11. GetLastGenICamError method

This method retrieves information of GenICamError.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetLastGenICamError(  
    out string errorMessage  
)
```

[VB.NET]

```
Public Function GetLastGenICamError (  
    <OutAttribute> ByRef errorMessage As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetLastGenICamError(  
    [OutAttribute] String^% errorMessage  
)
```

#### [Parameters]

Parameters	Description
<i>errorMessage</i>	A variable that receives retrieved error information.

#### [Return value]

Returns result status.

#### [Remarks]

GenICamError is an error that may occur during executing methods of [GenApiWrapper](#) class or [CameraControl](#) class, that will return `CamApiStatus.GenICamError`.

TeliCamDNetAPI collects error information of all threads into an error information variable.

The received error information may different from the desired error information, when another GenICamError occurred in another thread before or during executing this method.

---

## 5.2.2. Member Variables (Fields)

### 5.2.2.1. camStream field

This variable is a reference to [CameraStream](#) object.

Use this variable for receiving images.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CameraStream camStream
```

[VB.NET]

```
Public camStream As CameraStream
```

[C++]

```
public:  
CameraStream^ camStream
```

---

**[Remarks]**

This object will be created on creating [CameraDevice](#) class instance, and is under the management of [CameraDevice](#) class.

User application should not dispose (delete) this object.

---

#### 5.2.2.2. `camEvent` field

This variable is a reference to [CameraEvent](#) object.

Use this variable for receiving camera-events.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public CameraEvent camEvent
```

[VB.NET]

```
Public camEvent As CameraEvent
```

[C++]

```
public:  
CameraEvent^ camEvent
```

##### **[Remarks]**

---

This object will be created on creating [CameraDevice](#) class instance, and is under the management of [CameraDevice](#) class.

User application should not dispose (delete) this object.

---

### 5.2.2.3. [camControl](#) field

This variable is a reference to [CameraControl](#) object .

Use this variable for controlling camera features.

namespace: Teli.TeliCamAPI.NET

---

#### **[Syntax]**

[C#]

```
public CameraControl camControl
```

[VB.NET]

```
Public camControl As CameraControl
```

[C++]

```
public:  
CameraControl ^ camControl
```

---

#### **[Remarks]**

This object will be created on creating [CameraDevice](#) class instance, and is under the management of [CameraDevice](#) class.

User application should not dispose (delete) this object.



---

#### 5.2.2.4. [genApi](#) field

This variable is a reference to [GenApiWrapper](#) object.

Use this variable for controlling camera features including features that [CameraControl](#) class does not support.

Most cameras contain GenICam GenApi compliant camera description file (XML file).

XML file contains information of camera registers including register address, type of register data, accessibility, and many other informations associating with name assigned to the feature of the register, etc., except value of registers.

[GenApiWrapper](#) object provides the way to access camera registers using name assigned to the feature of the register, without specifying register address, utilizing information in the XML file.

namespace: Teli.TeliCamAPI.NET

---

##### [Syntax]

[C#]

```
public GenApiWrapper genApi
```

[VB.NET]

```
Public genApi As GenApiWrapper
```

[C++]

```
public:  
GenApiWrapper^ genApi
```

---

##### [Remarks]

This object will be created on creating [CameraDevice](#) class instance, and is under the management of [CameraDevice](#) class.

User application should not dispose (delete) this object.

---

## 5.2.3. Properties

### 5.2.3.1. CamType property

This property gets interface type of the camera.

CameraType.Unknown” will be returned when the camera is not opened yet.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CameraType CamType { get; }
```

[VB.NET]

```
Public ReadOnly Property CamType As CameraType  
Get
```

[C++]

```
public:  
property CameraType CamType {  
    CameraType get ();  
}
```

---

### 5.2.3.2. CamIndex property

This property gets index of the camera at the timing when CameraDevice object was created.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public int CamIndex { get; }
```

[VB.NET]

```
Public ReadOnly Property CamIndex As Integer  
    Get
```

[C++]

```
public:  
property Int32 CamIndex {  
    Int32 get ();  
}
```

#### **[Remarks]**

---

The latest camera index may different from this property value, if the camera list inside TeliCamDNetAPI (TeliCamAPI) has been updated by executing "[GetNumOfCameras\(\)](#)" method.

---

### 5.2.3.3. IsOpen property

This property gets a value indicating whether the camera is opened.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public bool IsOpen { get; }
```

[VB.NET]

```
Public ReadOnly Property IsOpen As Boolean  
    Get
```

[C++]

```
public:  
property Boolean IsOpen {  
    Boolean get ();  
}
```

#### **[Remarks]**

---

This property will return true when the camera is opened.  
Otherwise, this property will return false.

---

## 5.2.4. Events

### 5.2.4.1. Removed event

“Removed” event occurs when the camera is disconnected.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public event RemovedEventHandler Removed
```

[VB.NET]

```
Public Event Removed As RemovedEventHandler
```

[C++]

```
public:  
event RemovedEventHandler^ Removed {  
    void add (RemovedEventHandler^ value);  
    void remove (RemovedEventHandler^ value);  
}
```

---

**[Remarks]**

This event occurs when device driver detects disconnection of the camera.

Device driver will detect disconnection when the camera is disconnected, when the camera is reset, or when “[ResetPort\(\)](#)” method is executed, etc.

In USB3 Vision camera case, this event occurs when device driver detects disconnection of the camera.

In GigE Vision camera case, this event occurs when TeliCamDNetAPI (TeliCamAPI) detects Heartbeat timeout. The default timeout period is 15 seconds. This event will not occur if Heartbeat is disabled.

---

## 5.3. CameraStream class

CameraStream is a class for controlling a (image) stream.

CameraStream uses internal ImageRingBuffer as a buffer between device driver and user application. User application can retrieve image data from the internal ImageRingBuffer.

CameraStream provides ImageAcquired, ImageErrorReceived, and BufferBusyReceived events for making source code for acquiring image simple.

Refer to section “4.1.5 Streaming control” in [“TeliCamAPI Library Manual Eng.pdf”](#) about streaming.

CameraStream object will be created as a member of [CameraDevice](#) object on creating [CameraDevice](#) class instance, and is under the management of [CameraDevice](#) class.

User application should not dispose (delete) CameraStream object.

namespace: Teli.TeliCamAPI.NET

---

### [Syntax]

[C#]

```
public class CameraStream : IDisposable
```

[VB.NET]

```
Public Class CameraStream  
    Implements IDisposable
```














[C++]

```
public ref class CameraStream : IDisposable
```


The followings are members of CameraStream class available for user applications.

---

### [Methods]


	Name	Description
	Open(Int32%)	Opens a stream interface for receiving images.
	Open(AutoResetEvent, Int32, Int32, Int32%)	Opens a stream interface for receiving image with parameters.
	Close	Closes the stream interface.
	Start()	Sends streaming start command to the camera.
	Start(CameraAcquisitionMode)	Sends streaming start command to the camera specifying acquisition mode.
	Stop	Sends streaming stop command to the camera.
	Abort	Sends streaming abort command to the camera.
	ReadCurrentImage(IntPtr, Int32, CameraImageInfo )	Copies the latest image data in ImageRingBuffer to a memory block.
	ReadCurrentImage(Byte[], Int32, CameraImageInfo)	Copies the latest image data in ImageRingBuffer to byte array variable.
	GetCurrentBufferIndex	Gets index of buffer in ImageRingBuffer that contains the latest StreamRequest (image data).
	LockBuffer	Locks the buffer of the specified index in ImageRingBuffer to get image information in it.
	UnlockBuffer	Unlocks the buffer of the specified index in ImageRingBuffer.
	ChunkAttachBuffer(IntPtr,	Attach the buffer to the GenICam chunk port.

---

	UInt32)	
	ChunkAttachBuffer(Byte[] , UInt32)	Attach the buffer to the GenICam chunk port.




#### [Properties]

---

	Name	Description
	Parent	Gets parent <a href="#">CameraDevice</a> object.

#### [Events]

---

	Name	Description
	ImageAcquired	Occurs when stream (image) data has been received successfully.
	ImageErrorReceived	Occurs when failed in receiving stream (image) data.
	BufferBusyReceived	Occurs when the received stream (image) data is discarded. This event will occur when a buffer, that the received stream data is going to be written, is being locked by user application.

---

### 5.3.1. Methods

#### 5.3.1.1. Open method

This method opens a stream interface for receiving images.

This method also creates ImageRingBuffer in TeliCamDNetAPI (TeliCamAPI) to store received stream (image) data temporary.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public CamApiStatus Open(  
    out int maxPayloadSize  
)  
  
public CamApiStatus Open(  
    AutoResetEvent receivedEvent,  
    int apiBufferCount,  
    int maxPacketSize,  
    out int maxPayloadSize  
)
```

[VB.NET]

```
Public Function Open (  
    <OutAttribute> ByRef maxPayloadSize As Integer  
) As CamApiStatus  
  
Public Function Open (  
    receivedEvent As AutoResetEvent,  
    apiBufferCount As Integer,  
    maxPacketSize As Integer,  
    <OutAttribute> ByRef maxPayloadSize As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Open(  
    [OutAttribute] Int32% maxPayloadSize  
)  
  
public:  
CamApiStatus Open(  
    AutoResetEvent^ receivedEvent,  
    Int32 apiBufferCount,  
    Int32 maxPacketSize,  
    [OutAttribute] Int32% maxPayloadSize  
)
```



---

**[Parameters]**

Parameters	Description
<i>maxPayloadSize</i>	<p>A variable that contains the maximum payload size (image size in bytes), which is written to image buffer in a <code>StreamRequest</code>.</p> <p>The image size and <a href="#">PixelFormat</a> affects payload size.</p> <p>“<a href="#">GetStreamPayloadSize()</a>” will help you to get the default value for this parameter.</p>
<i>receivedEvent</i>	<p><code>AutoResetEvent</code> object for notifying that new image(s) has been received.</p> <p>Specify null (C#: null, VB.NET: Nothing, C++: nullptr) if notification is not necessary.</p>
<i>apiBufferCount</i>	<p>Number of <code>StreamRequest</code> (image) in <code>ImageRingBuffer</code>.</p> <p>The available value range is from 1 up to 128.</p> <p>If 0 is specified, default value (8) will be used as this parameter.</p>
<i>maxPacketSize</i>	<p>The maximum packet size that camera driver can receive, in bytes. This argument is optional.</p> <p>If 0 is specified, or value is not specified, the following default value will be used.</p> <p>USB3 Vision camera : 65536 bytes GigE Vision camera : 1500 bytes</p> <p>When decreasing overhead of streaming is required, enable Jumbo-Frame of network card that GigE Vision camera is connected, and specify the Jumbo-Frame size to <code>maxPacketSize</code> parameter.</p> <p>Note that <code>maxPacketSize</code> value is packet size excluding Ethernet header (14 bytes). If the network card uses packet size value including Ethernet header as Jumbo-Frame size, specify (Jumbo-Frame size – 14).</p> <p>For example, when Jumbo-Frame size is 9014, which usually includes Ethernet header size, specify 9000 instead of 9014.</p> <p>Specify 0 or do not specify value for USB3 Vision camera.</p>

---

**[Return value]**

Returns result status.

---

**[Remarks]**

If user application calls this method for a `CameraStream` object that the other application is using, this method will return `CamApiStatus.AlreadyOpened`.

When `CameraStream` object has finished saving an image data to a buffer in the `ImageRingBuffer`, `CameraStream` object will set “[receivedEvent](#)” signaled, then raise “[ImageAcquired](#)” event.

`CameraStream` class provides three ways for getting received image in the `ImageRingBuffer`,

- 
1. Using event handler registered to "[ImageAcquired](#)" event.

The "ImageAcquired" event handler can get image data and information of the image as parameters itself. Note that a stream request in the ImageRingBuffer that contains the latest image data is locked during "ImageAcquired" event handler is running.

2. Using "[ReadCurrentImage\(\)](#)" method.

"[ReadCurrentImage\(\)](#)" method copies the latest image data to the specified buffer and informs reference to information of the latest image.

3. Using "[GetCurrentBufferIndex\(\)](#)", "[LockBuffer\(\)](#)", and "[UnlockBuffer\(\)](#)" methods.

User application can get image in any index of the ImageRingBuffer. The followings shows steps for getting an image data.

- A. Call "[GetCurrentBufferIndex\(\)](#)" to get index of the StreamRequest that contains current image in the ImageRingBuffer.
- B. Calculate index of target StreamRequest using index of current Streamrequest.
- C. Call "[LockBuffer\(\)](#)" to lock the target StreamRequest .
- D. Get image data and its information using data returned to "[LockBuffer\(\)](#)" arguments.
- E. Call "[UnlockBuffer\(\)](#)" to unlock target Streamrequest.

If the higher priority threads are running, plural frames of newly received image may have been saved to ImageRingBuffer before a processig starts which is activated by "*receivedEvent*" or registered to "[ImageAcquired](#)" event handler.

Note that "[receivedEvent](#)" will not be signaled again corresponding to the already saved image, "[ImageAcquired](#)" event will not be raised again corresponding to the already saved image.

When plural frames of newly saved image may have been saved to ImageRingBuffer before processing of the the previous "[ImageAcquired](#)" event handler is done, "[ImageAcquired](#)" event will be raised once for the plural images saved to ImageRingBuffer.

When processing all images is required, use "[GetCurrentBufferIndex\(\)](#)" method for checking received frame count, and use "[LockBuffer\(\)](#)", and "[UnlockBuffer\(\)](#)" methods for processing images not processed yet.

Note that ImageRingBuffer in TeliCamDNetApi(TeliCamAPI) is not image buffer in the camera.

---

### 5.3.1.2. Close method

This method closes the stream interface for receiving images.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public CamApiStatus Close()
```

[VB.NET]

```
Public Function Close As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Close()
```

#### **[Return value]**

---

Returns result status.

#### **[Remarks]**

---

User application has to close the opened CameraStream object using "Close()" method when finished using the CameraStream object.

Calling this method during the other thread is using the CameraStream object may cause error in the other thread.

Call this method after all threads finished using the CameraStream.

---

### 5.3.1.3. Start method

This method sends streaming start command to the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus Start()  
  
public CamApiStatus Start(  
    CameraAcquisitionMode mode  
)
```

[VB.NET]

```
Public Function Start As CamApiStatus  
  
Public Function Start (  
    mode As CameraAcquisitionMode  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Start()  
  
public:  
CamApiStatus Start(  
    CameraAcquisitionMode mode  
)
```

#### [Parameters]

---

Parameters	Description
<i>mode</i>	Image acquisition mode. “ <a href="#">CameraAcquisitionMode</a> .Continuous” (continuous streaming mode) is specified to mode parameter in case of “Start()” method with no arguments.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

The following table describes about each [image acquisition mode](#).

Refer to instruction manual of the camera about the available [image acquisition mode](#).

CameraAcquisitionMode	Description
Continuous	Camera repeats acquiring images and sending them as stream until “ <a href="#">Stop()</a> ” is called.

---

CameraAcquisitionMode	Description
MultiFrame	<p>Camera repeats acquiring images and sending them as stream until a number of images specified by "<a href="#">SetAcquisitionFrameCount()</a>" method are acquired.</p> <p>Calling "<a href="#">Stop()</a>" is required even if streaming was stopped after the specified number of images were acquired.</p>
ImageBufferRead	<p>Camera repeats acquiring images and saving them to buffers in the camera until "<a href="#">Stop()</a>" is called.</p> <p>User application can instruct camera to send image in the camera buffer using "<a href="#">ExecuteImageBufferRead()</a>".</p>

Set image buffer mode ON using "[SetImageBufferMode\(\)](#)" before calling "Start()" with CameraAcquisitionMode.ImageBufferRead, otherwise set image buffer mode Off.

"TLParamsLocked" will be set to 1 when this function is called, if GenICam function is enabled.

---

#### 5.3.1.4. Stop method

This method sends streaming stop command to the camera.

This function stops the acquisition of the camera at the end of the current Frame.

namespace: Teli.TeliCamAPI.NET

---

##### **[Syntax]**

[C#]

```
public CamApiStatus Stop()
```

[VB.NET]

```
Public Function Stop As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Stop()
```

---

##### **[Return value]**

Returns result status.

---

##### **[Remarks]**

“TLParamsLocked” will be set to 0 when this function is called, if GenICam function is enabled.

---

### 5.3.1.5. Abort method

This method sends streaming abort command to the camera.

This function abort the acquisition immediately.

This will end the capture without completing the current Frame.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public CamApiStatus Abort()
```

[VB.NET]

```
Public Function Abort As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Abort()
```

#### **[Return value]**

---

Returns result status.

#### **[Remarks]**

---

The aborted frame is given an error status.

If the frame is aborted without completing, [ImageErrorReceived](#) event is called.

“TLParamsLocked” will be set to 0 when this function is called, if GenICam function is enabled.

---

### 5.3.1.6. ReadCurrentImage methods

This method obtains the latest image data in ImageRingBuffer to a memory block.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus ReadCurrentImage(  
    IntPtr buffer,  
    ref int size,  
    ref CameraImageInfo imageInfo  
)
```

```
public CamApiStatus ReadCurrentImage(  
    byte[] byteArray,  
    out int copySize,  
    ref CameraImageInfo imageInfo  
)
```

[VB.NET]

```
Public Function ReadCurrentImage (  
    buffer As IntPtr,  
    ByRef size As Integer,  
    ByRef imageInfo As CameraImageInfo  
) As CamApiStatus
```

```
Public Function ReadCurrentImage (  
    byteArray As Byte(),  
    <OutAttribute> ByRef copySize As Integer,  
    ByRef imageInfo As CameraImageInfo  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus ReadCurrentImage(  
    IntPtr buffer,  
    Int32% size,  
    CameraImageInfo^% imageInfo  
)
```

```
public:  
CamApiStatus ReadCurrentImage(  
    array<Byte>^ byteArray,  
    [OutAttribute] Int32% copySize,  
    CameraImageInfo^% imageInfo  
)
```



---

**[Parameters]**

---

Parameters	Description
<i>buffer</i>	An image data block that receives the latest image.
<i>size</i>	<p>A variable that contains size of image data block.</p> <p>If size of image data block is less than size of the received image, error status will be returned.</p> <p>The size of copied image in bytes will be returned to this variable after copying image data.</p>
<i>imageInfo</i>	A variable that receives additional information of the latest image.
<i>buffer</i>	A byte array that receives the latest image.
<i>copySize</i>	Variable that receives the size of copied image in bytes.

**[Return value]**

---

Returns result status. If error occurred during acquiring images, error code will be returned.

---

### 5.3.1.7. GetCurrentBufferIndex method

This method gets index of buffer in ImageRingBuffer that contains the latest StreamRequest (image data).

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetCurrentBufferIndex(  
    out int bufferIndex  
)
```

[VB.NET]

```
Public Function GetCurrentBufferIndex (  
    <OutAttribute> ByRef bufferIndex As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetCurrentBufferIndex(  
    [OutAttribute] Int32% bufferIndex  
)
```

#### [Parameters]

---

Parameters	Description
<i>bufferIndex</i>	<p>A variable that receives index of current StreamRequest (image).</p> <p>The index will be from 0 up to the number specified as "<a href="#">apiBufferCount</a>" parameter of "<a href="#">Open()</a>" method minus 1.</p> <p>If no images have been stored to ImageRingBuffer, 0xFFFFFFFF will be written as buffer index.</p>

#### [Return value]

---

Returns result status.

---

### 5.3.1.8. LockBuffer method

This method locks the buffer of the specified index in ImageRingBuffer to get image information in it.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus LockBuffer(  
    int bufferIndex,  
    ref CameraImageInfo imageInfo  
)
```

[VB.NET]

```
Public Function LockBuffer (  
    bufferIndex As Integer,  
    ByRef imageInfo As CameraImageInfo  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus LockBuffer(  
    Int32 bufferIndex,  
    CameraImageInfo^% imageInfo  
)
```

#### [Parameters]

Parameters	Description
<i>bufferIndex</i>	Index of the StreamRequest (image) to lock. The available argument value is from 0 up to the number specified as " <a href="#">apiBufferCount</a> " parameter of " <a href="#">Open()</a> " method minus 1.
<i>imageInfo</i>	A variable that receives information of the image.

#### [Return value]

Returns result status.

#### [Remarks]

The StreamRequest locked using this function should be unlocked using "[UnlockBuffer\(\)](#)" as soon as possible.

Images received from the camera will be stored to StreamRequests (structure which contains image data) in the fixed order. If a StreamRequest that the receiving image should be written is left locked, the receiving image will be discarded and "[BufferBusyReceived](#)" event will be raised.

To avoid discarding received image due to buffer busy error, shorten processing contents performed during a StreamRequest is locked, or set the larger value to "[apiBufferCount](#)" on calling "[Open\(\)](#)".

---

### 5.3.1.9. UnlockBuffer method

This method unlocks the buffer of the specified index in ImageRingBuffer.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus UnlockBuffer(  
    int bufferIndex  
)
```

[VB.NET]

```
Public Function UnlockBuffer (  
    bufferIndex As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus UnlockBuffer(  
    Int32 bufferIndex  
)
```

#### [Parameters]

Parameters	Description
<i>bufferIndex</i>	Index of the StreamRequest to unlock.  The available argument value is from 0 up to the number specified as " <a href="#">apiBufferCount</a> " parameter of " <a href="#">Open()</a> " minus 1.

#### [Return value]

Returns result status.

#### [Remarks]

The StreamRequest locked using "[LockBuffer\(\)](#)" should be unlocked using this function as soon as possible.

---

### 5.3.1.10. ChunkAttachBuffer method

This method attaches the buffer to the GenICam chunk port.

When acquiring chunk data from received payload data using GenICam, it is necessary to attach the buffer storing the payload data to GenICam chunk port.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus ChunkAttachBuffer (
    IntPtr buffer,
    uint payloadSize
)

public CamApiStatus ChunkAttachBuffer (
    byte[] byteArray,
    uint payloadSize
)
```

[VB.NET]

```
Public Function ChunkAttachBuffer (
    buffer As IntPtr,
    payloadSize As UInteger
) As CamApiStatus

Public Function ChunkAttachBuffer (
    byteArray As Byte(),
    payloadSize As UInteger
) As CamApiStatus
```

[C++]

```
public:
CamApiStatus ChunkAttachBuffer (
    IntPtr buffer,
    unsigned int payloadSize
)

public:
CamApiStatus ChunkAttachBuffer (
    array<Byte>^ byteArray,
    unsigned int payloadSize
)
```

---

**[Parameters]**

---

Parameters	Description
buffer	IntPtr to the buffer storing the received payload data.
payloadSize	The size of the received payload data.
byteArray	A byte type data array storing the received payload data.

**[Return value]**

---

Returns result status.

**[Remarks]**

---

This function will return error status if no chunk data is added to the payload data.

Chunk data is acquired using the GenApiWrapper class.

---

## 5.3.2. Properties

### 5.3.2.1. Parent property

This property gets parent [CameraDevice](#) object.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public CameraDevice Parent { get; }
```

[VB.NET]

```
Public ReadOnly Property Parent As CameraDevice  
    Get
```

[C++]

```
public:  
property CameraDevice^ Parent {  
    CameraDevice^ get ();  
}
```

---

### 5.3.3. Events

#### 5.3.3.1. ImageAcquired event

This event occurs when stream (image) data has been received successfully.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public event ImageAcquiredEventHandler ImageAcquired
```

[VB.NET]

```
Public Event ImageAcquired As ImageAcquiredEventHandler
```

[C++]

```
public:  
event ImageAcquiredEventHandler^ ImageAcquired {  
    void add (ImageAcquiredEventHandler^ value);  
    void remove (ImageAcquiredEventHandler^ value);  
}
```

---

**[Remarks]**

“ImageAcquired” event will be raised when the content of ImageRingBuffer has been updated, after receiving stream (image) data successfully.



---

### 5.3.3.2. ImageErrorReceived event

This event occurs when failed in receiving stream (image) data.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public event ImageErrorReceivedEventHandler ImageErrorReceived
```

[VB.NET]

```
Public Event ImageErrorReceived As ImageErrorReceivedEventHandler
```

[C++]

```
public:  
event ImageErrorReceivedEventHandler^ ImageErrorReceived {  
    void add (ImageErrorReceivedEventHandler^ value);  
    void remove (ImageErrorReceivedEventHandler^ value);  
}
```

#### [Remarks]

---

“ImageErrorReceived” event will be raised when the content of ImageRingBuffer has been updated with stream data containing error, due to failure in receiving stream (image) data.

---

### 5.3.3.3. BufferBusyReceived event

This event occurs when the received stream (image) data is discarded.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public event BufferBusyReceivedEventHandler BufferBusyReceived
```

[VB.NET]

```
Public Event BufferBusyReceived As BufferBusyReceivedEventHandler
```

[C++]

```
public:  
event BufferBusyReceivedEventHandler^ BufferBusyReceived {  
    void add (BufferBusyReceivedEventHandler^ value);  
    void remove (BufferBusyReceivedEventHandler^ value);  
}
```

#### [Remarks]

---

“BufferBusyReceived” event will be raised when a buffer, that the received stream data is going to be written, is being locked by user application.

---

## 5.4. CameraEvent class

CameraEvent is a class for controlling camera-events of a camera.

CameraEvent class provides events that correspond to camera-events of camera individually. CameraEvent object will execute methods registered to the eventhandler, on reception of the corresponding camera-events.

Methods of this class are not available when GenICam function was disabled on opening the parent CameraEvent object.

CameraEvent object will be created as a member of [CameraDevice](#) object on creating [CameraDevice](#) class instance, and is under the management of [CameraDevice](#) class.

User application should not dispose (delete) CameraEvent object.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

[C#]

```
public class CameraEvent : IDisposable
```

[VB.NET]







```
Public Class CameraEvent  
    Implements IDisposable
```

[C++]


```
public ref class CameraEvent : IDisposable
```

The followings are members of CameraEvent class available for user applications.

### [Methods]

	Name	Description
	Open()	Opens an event interface for receiving camera-event (message).
	Open(Int32)	Opens an event interface for receiving camera-event (message) with a paramer.
	Close	Close the event interface.
	Activate(CameraEventType)	Activates a specified camera-event.
	Activate(CameraEventType, AutoResetEvent)	Activates a specified camera-event and register AutoResetEvent object to the CameraEvent.
	Deactivate	Deactivates specified camera-event.












### [Properties]

	Name	Description
	Parent	Gets parent <a href="#">CameraDevice</a> object.

---

**[Events]**

---

	Name	Description
	FrameTriggerReceived	Occurs when the camera accepted trigger signal for acquiring an image.
	FrameTriggerErrorReceived	Occurs when the camera received trigger signal for acquiring image at invalid timing.
	FrameTriggerWaitReceived	Occurs when the camera started accepting trigger signal for image acquisition.
	FrameTransferStartReceived	Occurs when the camera started transferring image stream of a frame.
	FrameTransferEndReceived	Occurs when the camera completed transferring image stream of a frame.
	ExposureStartReceived	Occurs when the camera started exposure for a frame.
	ExposureEndReceived	Occurs when the camera finished exposure for a frame.
	Timer0StartReceived	Occurs when the camera started counting Timer0.
	Timer0EndReceived	Occurs when the camera finished counting Timer0.
	ALCLatestInformationReceived	Occurs when the camera updated Automatic Luminance Control data.
	ALCConvergedReceived	Occurs when Automatic Luminance Control operation of the camera has been converged.

---

## 5.4.1. Methods

### 5.4.1.1. Open method

This method opens an event interface for receiving camera-events (message).

This method also creates EventRingBuffer in TeliCamDNetAPI (TeliCamAPI) to store received camera-event (message) data temporary.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus Open()  
  
public CamApiStatus Open(  
    int apiBufferCount  
)
```

[VB.NET]

```
Public Function Open As CamApiStatus  
  
Public Function Open (  
    apiBufferCount As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Open()  
  
public:  
CamApiStatus Open(  
    Int32 apiBufferCount  
)
```

#### [Parameters]

Parameters	Description
apiBufferCount	Size (number of EventRequest) of the EventRingBuffer for the opening event interface.  The available value range is from 3 up to 30.  If 0 is specified, default value (8) will be used as this parameter.

#### [Return value]

Returns result status.

#### [Remarks]

If user application calls this method for a CameraEvent object that the other application is using, this method will return `CamApiStatus.AlreadyOpened`.

---

Activating plural camera-event of a camera will make response for camera-event notifications worse. The incoming EventRequest may be silently discarded when no room for accepting the incoming camera-event data are left in EventRingBuffer, if processing cost of user application event handler is heavy or CPU power is not enough.

No error will be reported even if the incoming EventRequest is discarded.

We recommend users to refrain from activating plural camera-event when acquiring images continuously.

---

#### 5.4.1.2. Close method

This method closes the event interface.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

[C#]

```
public CamApiStatus Close()
```

[VB.NET]

```
Public Function Close As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Close()
```

##### **[Return value]**

Returns result status.

##### **[Remarks]**

User application has to close the opened CameraEvent object using "Close()" method when finished using the CameraEvent object.

Never call "Close()" method during the other method is using the CameraEvent object.

---

### 5.4.1.3. Activate method

This method activates a specified camera-event.

This method also register AutoResetEvent object if it is specified as parameter of this method.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus Activate(  
    CameraEventType eventType  
)  
  
public CamApiStatus Activate(  
    CameraEventType eventType,  
    AutoResetEvent receivedEvent  
)
```

[VB.NET]

```
Public Function Activate (  
    eventType As CameraEventType  
) As CamApiStatus  
  
Public Function Activate (  
    eventType As CameraEventType,  
    receivedEvent As AutoResetEvent  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus Activate(  
    CameraEventType eventType  
)  
  
public:  
CamApiStatus Activate(  
    CameraEventType eventType,  
    AutoResetEvent^ receivedEvent  
)
```

#### [Parameters]

---

Parameters	Description
<i>eventType</i>	camera-event to activate.
<i>receivedEvent</i>	AutoResetEvent object for notifying reception of the camera-event. Specify null (C#:null, VB.Net:Nothing, C++/CLI:nullptr) if notification is not required.

#### [Return value]

---

Returns result status.



---

## [Remarks]

CameraEvent class provides the following two types of cameraevent notifications.

- Using event handler registered to CameraEvent class event.  
CameraEvent class provides events that correspond to camera-events of camera individually. CameraEvent object will execute registered event handler on reception of the corresponding camera-events.

- Using AutoResetEvent object.

This method registers "[receivedEvent](#)" AutoResetEvent object to the CameraEvent object.

The registered AutoResetEvent object will be signaled on reception of camera-event notification from the camera.

If CameraEvent object received a camera-event again before user application sets corresponding AutoResetEvent object not signaled, CameraEvent object will set the AutoResetEvent object signaled without waiting that the AutoResetEvent object is reset by user application.

User application can get information of camera-event using [GenApiWrapper](#) class methods, after receiving the camera-event.

The following table shows node name of event information.

CameraEventTypee	Node name of event information
FrameTrigger	EventFrameTriggerTimestamp
FrameTriggerError	EventFrameTriggerErrorTimestamp
FrameTriggerWait	EventFrameTriggerWaitTimestamp
FrameTransferStart	EventFrameTransferStartTimestamp
FrameTransferEnd	EventFrameTransferEndTimestamp
ExposureStart	EventExposureStartTimestamp
ExposureEnd	EventExposureEndTimestamp
Timer0Start	EventTimer0StartTimestamp
Timer0End	EventTimer0EndTimestamp
ALCLatestInformation	EventALCLatestInformationTimestamp
	EventALCLatestInformationTotalLuminance
	EventALCLatestInformationAverageLuminance
	EventALCLatestInformationExposureTime
	EventALCLatestInformationGain
ALCConverged	EventALCConvergedTimestamp
	EventALCConvergedLuminanceTotal
	EventALCConvergedLuminanceAverage
	EventALCConvergedExposureTime
	EventALCConvergedGain

Refrain from accessing to node whose node name is name of CameraEventType with prefix "Event" using [GenApiWrapper](#) class method ("[GetNode\(\)](#)"). Accessing to the node will set "[receivedEvent](#)" signaled and the method registered to corresponding eventhandler will be executed.

There are various restrictions in using camera event. For example, setting parameters in a wrong order may cause or FrameTrigger cannot be activated when TriggerMode of the camera is off, and so on.

Refer to instruction manual of the camera.

The following figures indicate typical timing of camera-events.

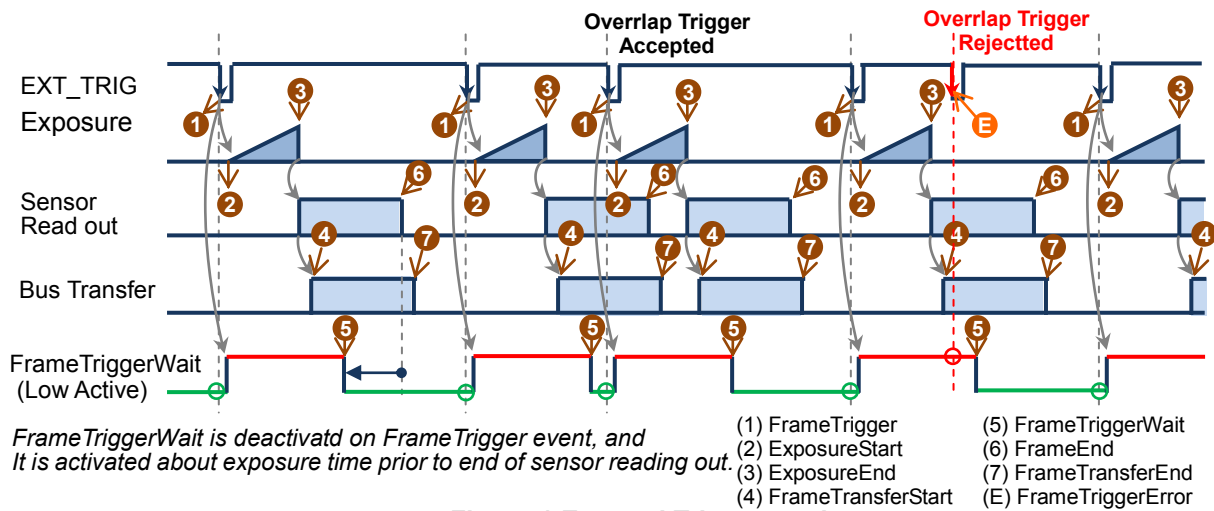


Figure 1 External Trigger mode

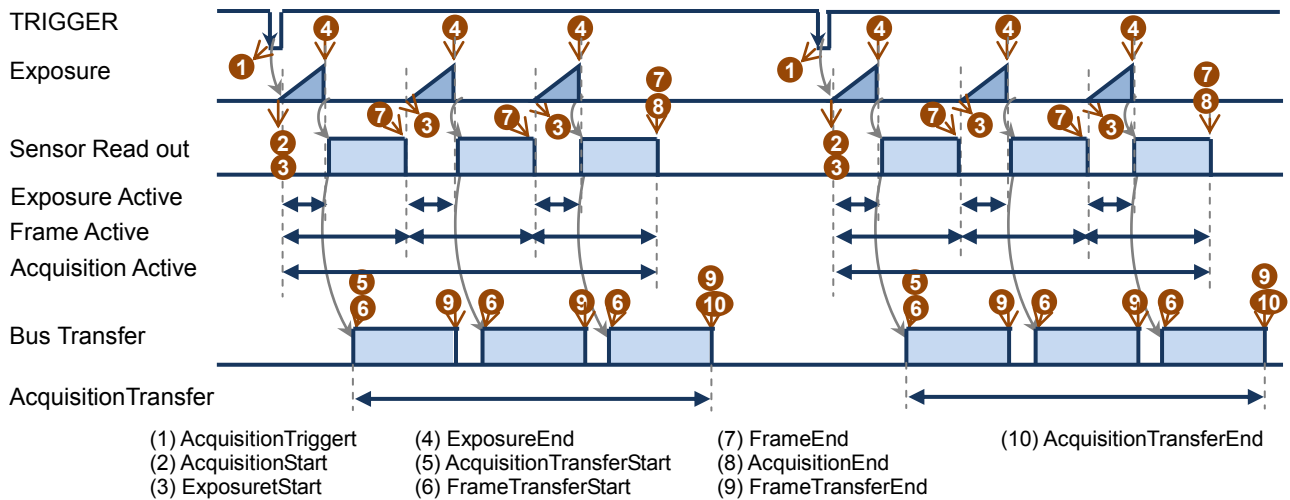


Figure 2 Bulk Trigger mode, 3 frames per a trigger

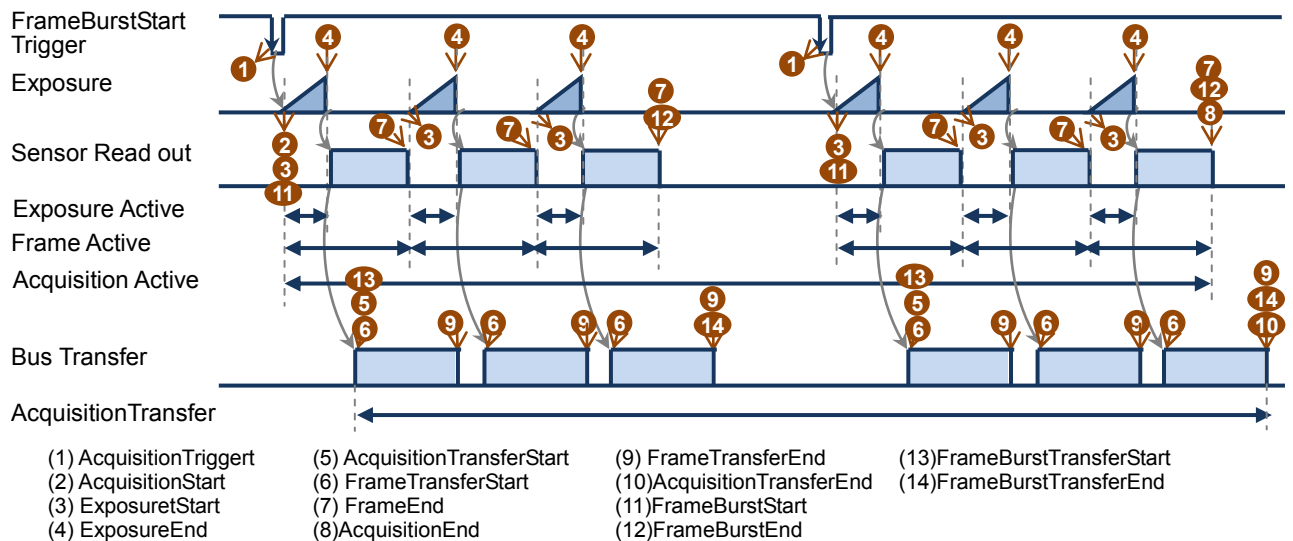


Figure 3 Burst Signals

---

#### 5.4.1.4. Deactivate method

This method deactivates specified camera-event.

This method also releases registration of AutoResetEvent object registered using "[Activate\(\)](#)" method.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public CamApiStatus Deactivate(  
    CameraEventType eventType  
)
```

[VB.NET]

```
Public Function Deactivate (  
    eventType As CameraEventType  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus Deactivate(  
        CameraEventType eventType  
)
```

##### [Parameters]

---

Parameters	Description
eventType	camera-event to deactivate.

##### [Return value]

---

Returns result status.

##### [Remarks]

---

The camera event that activated using "[Activate\(\)](#)" must be released using this function before closing the event interface.

---

## 5.4.2. Properties

### 5.4.2.1. Parent property

This property gets parent [CameraDevice](#) object.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CameraDevice Parent { get; }
```

[VB.NET]

```
Public ReadOnly Property Parent As CameraDevice  
    Get
```

[C++]

```
public:  
property CameraDevice^ Parent {  
    CameraDevice^ get ();  
}
```

---

### 5.4.3. Events

#### 5.4.3.1. FrameTriggerReceived event

This event occurs when the camera accepted trigger signal for acquiring an image.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public event CameraEventReceivedEventHandler FrameTriggerReceived
```

[VB.NET]

```
Public Event FrameTriggerReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ FrameTriggerReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```

#### 5.4.3.2. FrameTriggerErrorReceived event

This event occurs when the camera received trigger signal for acquiring image at invalid timing.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public event CameraEventReceivedEventHandler FrameTriggerErrorReceived
```

[VB.NET]

```
Public Event FrameTriggerErrorReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ FrameTriggerErrorReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```

---

#### 5.4.3.3. FrameTriggerWaitReceived event

This event occurs when the camera started accepting trigger signal for image acquisition.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public event CameraEventReceivedEventHandler FrameTriggerWaitReceived
```

[VB.NET]

```
Public Event FrameTriggerWaitReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ FrameTriggerWaitReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```

#### 5.4.3.4. FrameTransferStartReceived event

This event occurs when the camera started transferring image stream of a frame

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public event CameraEventReceivedEventHandler FrameTransferStartReceived
```

[VB.NET]

```
Public Event FrameTransferStartReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ FrameTransferStartReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```

---

#### 5.4.3.5. FrameTransferEndReceived event

This event occurs when the camera completed transferring image stream of a frame.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public event CameraEventReceivedEventHandler FrameTransfrEndReceived
```

[VB.NET]

```
Public Event FrameTransferEndReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ FrameTransferEndReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```

#### 5.4.3.6. ExposureStartReceived event

This event occurs when the camera started exposure for a frame.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public event CameraEventReceivedEventHandler ExposureStartReceived
```

[VB.NET]

```
Public Event ExposureStartReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ ExposureStartReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```

---

#### 5.4.3.7. ExposureEndReceived event

This event occurs when the camera finished exposure for a frame.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public event CameraEventReceivedEventHandler ExposureEndReceived
```

[VB.NET]

```
Public Event ExposureEndReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ ExposureEndReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```

#### 5.4.3.8. Timer0StartReceived event

This event occurs when the camera started counting Timer0.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public event CameraEventReceivedEventHandler Timer0StartReceived
```

[VB.NET]

```
Public Event Timer0StartReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ Timer0StartReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```



---

#### 5.4.3.9. Timer0EndReceived event

This event occurs when the camera finished counting Timer0.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public event CameraEventReceivedEventHandler Timer0EndReceived
```

[VB.NET]

```
Public Event Timer0EndReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ Timer0EndReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```

#### 5.4.3.10. ALCLatestInformationReceived event

This event occurs when the camera updated Automatic Luminance Control data.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public event CameraEventReceivedEventHandler ALCLatestInformationReceived
```

[VB.NET]

```
Public Event ALCLatestInformationReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ LCLatestInformationReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```

---

#### 5.4.3.11. ALCConvergedReceived event

This event occurs when Automatic Luminance Control operation of the camera has been converged.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public event CameraEventReceivedEventHandler ALCConvergedReceived
```

[VB.NET]

```
Public Event ALCConvergedReceived As CameraEventReceivedEventHandler
```

[C++]

```
public:  
event CameraEventReceivedEventHandler^ ALCConvergedReceived {  
    void add (CameraEventReceivedEventHandler^ value);  
    void remove (CameraEventReceivedEventHandler^ value);  
}
```

---

## 5.5. CameraControl class

CameraControl is a class for controlling features of a camera. CameraControl object allows user application to control a camera without paying attention to interface type, model of the camera, and address of registers.

In GigE Vision camera case, all methods in this class use GenICam GenApi library for controlling camera. In USB3 Vision camera, most methods in this class access camera register directly, which will realize the higher performance. The methods that use GenICam GenApi function are not available when GenApi module was disabled on opening the parent [CameraDevice](#) object.

There are functions that are available in all cameras. There also are functions that are not available in some model of camera because the corresponding features are not supported. Please check instruction manual of the camera to confirm the availability of the feature.

CameraControl object will be created as a member of [CameraDevice](#) object on creating [CameraDevice](#) class instance, and is under the management of [CameraDevice](#) class.

User application should not dispose (delete) CameraControl object.

Namespace: Teli.TeliCamAPI.NET

---

### [Syntax]

[C#]

```
Public class CameraControl: IDisposable
```

[VB.NET]

```
Public Class CameraControl  
    Implements IDisposable
```











[C++]
























```
Public ref class CameraControl: IDisposable
```

The followings are members of CameraControl class available for user applications.
































---





























### [Methods]










	Name	Description
	ImageFormatControl	
	GetImageFormatSelector	Gets image format of the camera.
	SetImageFormatSelector	Sets Image format of the camera.
	Scalable	
	GetSensorWidth	Gets number of active pixels in horizontal directions.
	GetSensorHeight	Gets number of active pixels in vertical directions.
	GetRoi	Gets <u>Region Of Interest</u> of the camera.
	SetRoi	Sets <u>Region Of Interest</u> of the camera.
	GetWidthMinMax	Gets the maximum and minimum available image width and increment value of the image width.
	GetWidth	Gets current image width in pixels.
	SetWidth	Sets image width in pixels.
	GetHeightMinMax	Gets the maximum and minimum available image height and increment value of the image height.

	Name	Description
	GetHeight	Gets current image height in pixels.
	SetHeight	Sets image height in pixels.
	GetOffsetXMinMax	Gets the maximum and minimum available horizontal image offset and increment value of image offset.
	GetOffsetX	Gets current horizontal image offset in pixels.
	SetOffsetX	Sets horizontal image offset in pixels.
	GetOffsetYMinMax	Gets the maximum and minimum available vertical image offset and increment value of image offset.
	GetOffsetY	Gets current vertical image offset in pixels.
	SetOffsetY	Sets vertical image offset in pixels.
<b>Binning</b>		
	GetBinningHorizontalMinMax	Gets the maximum and minimum available horizontal binning mode value.
	GetBinningHorizontal	Gets current horizontal binning mode value.
	SetBinningHorizontal	Sets horizontal binning mode value.
	GetBinningVerticalMinMax	Gets the maximum and minimum available vertical binning mode value.
	GetBinningVertical	Gets current vertical binning mode value.
	SetBinningVertical	Sets vertical binning mode value.
<b>Decimation</b>		
	GetDecimationHorizontalMinMax	Gets the maximum and minimum available horizontal decimation mode value.
	GetDecimationHorizontal	Gets current horizontal decimation mode value.
	SetDecimationHorizontal	Sets horizontal decimation mode value.
	GetDecimationVerticalMinMax	Gets the maximum and minimum available vertical decimation mode value.
	GetDecimationVertical	Gets current vertical decimation mode value.
	SetDecimationVertical	Sets vertical decimation mode value.
<b>Reverse</b>		
	GetReverseX	Gets current horizontal image reverse mode value.
	SetReverseX	Sets horizontal image reverse mode value.
	GetReverseY	Gets current vertical image reverse mode value.
	SetReverseY	Sets vertical image reverse mode value.
<b>PixelFormat</b>		
	GetPixelFormat	Gets PixelFormat of image stream.
	SetPixelFormat	Sets PixelFormat of image stream.
<b>TestPattern</b>		
	GetTestPattern	Gets test pattern mode value of the camera.
	SetTestPattern	Sets test pattern mode value of the camera.
<b>AcquisitionControl</b>		
	GetStreamPayloadSize	Gets payload size of image stream from the camera.
	GetStreamEnable	Gets streaming status (enabled / disabled) of the camera.
	GetAcquisitionFrameCountMinMax	Gets the maximum and minimum available "AcquisitionFrameCount" value.
	GetAcquisitionFrameCount	Gets current "AcquisitionFrameCount" value.


	Name	Description
⇒	SetAcquisitionFrameCount	Sets "AcquisitionFrameCount" value.
⇒	GetAcquisitionFrameRateControl	Gets current "AcquisitionFrameRateControl" register value.
⇒	SetAcquisitionFrameRateControl	Sets "AcquisitionFrameRateControl" register value.
⇒	GetAcquisitionFrameRateMinMax	Gets the maximum and minimum available frame rate value.
⇒	GetAcquisitionFrameRate	Gets current frame rate parameter.
⇒	SetAcquisitionFrameRate	Sets frame rate parameter.
ImageBuffer		
⇒	GetImageBufferMode	Gets current "ImageBufferMode" status.
⇒	SetImageBufferMode	Sets "ImageBufferMode" status.
⇒	GetImageBufferFrameCount	Gets number of images stored in image buffer of the camera.
⇒	ExecuteImageBufferRead	Sends command for reading out images from image buffer of the camera.
TriggerControl		
⇒	GetTriggerMode	Gets current triggering mode status.
⇒	SetTriggerMode	Sets triggering mode.
⇒	GetTriggerSequence	Gets current sequence of exposure time control.
⇒	SetTriggerSequence	Sets sequence of exposure time control.
⇒	GetTriggerSource	Gets current trigger source signal for random trigger shutter mode.
⇒	SetTriggerSource	Sets trigger source signal for random trigger shutter mode.
⇒	GetTriggerAdditionalParameterMinMax	Gets the maximum and minimum available value for "TriggerAdditionalParameter".
⇒	GetTriggerAdditionalParameter	Gets current "TriggerAdditionalParameter" value. This value is used as "number of frames acquired by a single trigger signal" in Bulk mode (TriggerSequence6)
⇒	SetTriggerAdditionalParameter	Sets "TriggerAdditionalParameter" value.
⇒	GetTriggerDelayMinMax	Gets the maximum and minimum value available for "TriggerDeley" register.
⇒	GetTriggerDelay	Gets current delay time from detection of trigger signal until starting exposure.
⇒	SetTriggerDelay	Sets delay time from detection of trigger signal until starting exposure.
⇒	ExecuteSoftwareTrigger	Sends software trigger command to the camera.
ExposureTime		
⇒	GetExposureTimeControl	Gets current exposure time control mode.
⇒	SetExposureTimeControl	Sets exposure time control mode.
⇒	GetExposureTimeMinMax	Gets the maximum and minimum available exposure time value.
⇒	GetExposureTime	Gets current exposure time.
⇒	SetExposureTime	Sets exposure time.

	Name	Description
<b>DigitalIOControl</b>		
	GetLineModeAll	Gets information of all I/O lines in the camera whether the line is input line or output line.
	SetLineModeAll	Sets information of all I/O lines in the camera whether the line is input line or output line.
	GetLineInverterAll	Gets current polarity of all I/O lines in the camera.
	SetLineInverterAll	Sets polarity of all I/O lines in the camera.
	GetLineStatusAll	Gets current status of all I/O lines in the camera.
	GetUserOutputValueAll	Gets current output value of all user output lines in the camera.
	SetUserOutputValueAll	Sets output value of all user output lines in the camera.
	GetLineSource	Gets current data source of the specified output line.
	SetLineSource	Sets data source of the specified output line.
<b>TimerControl</b>		
	GetTimerDurationMinMax	Gets the maximum and minimum available period that Timer0 signal is active.
	GetTimerDuration	Gets current period of Timer0 active.
	SetTimerDuration	Sets period of Timer0 active.
	GetTimerDelayMinMax	Gets the maximum and minimum available delay time for Timer0Active signal.
	GetTimerDelay	Gets current delay time for Timer0Active signal
	SetTimerDelay	Sets delay time for Timer0Active signal
	GetTimerTriggerSource	Gets current trigger source signal of Timer0.
	SetTimerTriggerSource	Sets trigger source signal of Timer0.
<b>Gain</b>		
	GetGainMinMax	Gets the maximum and minimum available gain value.
	GetGain	Gets current gain.
	SetGain	Sets current gain value.
	GetGainAuto	Gets current AGC mode.
	SetGainAuto	Sets AGC mode.
<b>BlackLevel</b>		
	GetBlackLevelMinMax	Gets the maximum and minimum available Black-Level value.
	GetBlackLevel	Gets current Black-Level.
	SetBlackLevel	Sets Black-Level value.
<b>Gamma</b>		
	GetGammaMinMax	Gets the maximum and minimum available gamma correction value.
	GetGamma	Gets current gamma correction value.
	SetGamma	Sets gamma correction value.
<b>WhiteBalance</b>		
	GetBalanceRatioMinMax	Gets the maximum and minimum available white balance gain value.
	GetBalanceRatio	Gets current white balance ratio.
	SetBalanceRatio	Sets white balance ratio.

	Name	Description
	GetBalanceWhiteAuto	Gets current automatic white balance mode.
	SetBalanceWhiteAuto	Sets automatic white balance mode.
Hue		
	GetHueMinMax	Gets the maximum and minimum available hue value.
	GetHue	Gets current hue correction value.
	SetHue	Sets hue correction value.
Saturation		
	GetSaturationMinMax	Gets the maximum and minimum available saturation value.
	GetSaturation	Gets current saturation value.
	SetSaturation	Sets saturation value.
Sharpness		
	GetSharpnessMinMax	Gets the maximum and minimum available sharpness feature value.
	GetSharpness	Gets current sharpness value.
	SetSharpness	Sets sharpness value.
ColorCorrectionMatrix		
	GetColorCorrectionMatrixMinMax	Gets the maximum and minimum coefficient of color correction matrix.
	GetColorCorrectionMatrix	Gets current coefficient of color correction matrix.
	SetColorCorrectionMatrix	Sets coefficient of color correction matrix.
LUT Control		
	GetLutEnable	Gets whether LUT feature of the camera is enabled.
	SetLutEnable	Enables or disables LUT feature of the camera
	GetLutValue	Get current value of LUT.
	SetLutValue	Set value to LUT.
UserSet Control		
	ExecuteUserSetLoad	Sends command for loading parameters stored in non- volatile UserSet memory to the camera.
	ExecuteUserSetSave	Sends command for saving current parameters in non- volatile UserSet memory to the camera.
	ExecuteUserSetQuickSave	Sends command for saving current parameters in volatile UserSet memory to the camera.
	GetUserSetDefault	Gets a user setting channel to be loaded when the camera is powered on.
	SetUserSetDefault	Sets a user setting channel to be loaded when the camera is powered on.ue.
	ExecuteUserSetSaveAndSetDefault	Sends command for saving current parameters in non- volatile UserSet memory and set it as default parameters of the camera.
SequentialShutterControl		
	GetSequentialShutterEnable	Gets current status of sequential shutter feature.
	SetSequentialShutterEnable	Enables / disables sequential shutter feature.
	GetSequentialShutter TerminateAtMinMax	Gets the maximum and minimum available value for "SequentialShutter" sequence length.
	GetSequentialShutterTerminateAt	Gets current "SequentialShutter" sequence length.

	Name	Description
	SetSequentialShutterTerminateAt	Sets "SequentialShutter" sequence length.
	GetSequentialShutterIndexMinMax	Gets the maximum and minimum available index of shutter definition data.
	GetSequentialShutterEntryMinMax	Gets the maximum and minimum UserSet channel number available for specifying shutter data.
	GetSequentialShutterEntry	Gets current UserSet channel number registered in a sequential shutter entry.
	SetSequentialShutterEntry	Registers UserSet channel number to sequential shutter entry.
UserDefinedName		
	GetUserDefinedName	Gets user defined name of the camera.
	SetUserDefinedName	Sets user defined name of the camera.
Chunk		
	GetChunkModeActive	Gets whether chunk data output mode is enabled.
	SetChunkModeActive	Enables or disables chunk data output mode.

#### [Properties]

	Name	Description
	Parent	Gets parent CameraDevice object.



---

### 5.5.1. ImageFormatControl methods

These methods control image format of the camera.  
Refer to instruction manual of the camera about image format.

#### 5.5.1.1. GetImageFormatSelector method

This method gets image format of the camera

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetImageFormatSelector(  
    out CameraImageFormat format  
)
```

[VB.NET]

```
Public Function GetImageFormatSelector (  
    <OutAttribute> ByRef format As CameraImageFormat  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetImageFormatSelector(  
    [OutAttribute] CameraImageFormat% format  
)
```

##### [Parameters]

Parameters	Description
<i>format</i>	A variable that receives image format.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "ImageFormatSelector" register or node is not implemented in the camera.

---

### 5.5.1.2. SetImageFormatSelector method

This method sets image format of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetImageFormatSelector(  
    CameraImageFormat format  
)
```

[VB.NET]

```
Public Function SetImageFormatSelector (  
    format As CameraImageFormat  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus SetImageFormatSelector(  
        CameraImageFormat format  
)
```

#### [Parameters]

Parameters	Description
<i>format</i>	New image format value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "ImageFormatSelector" register or node is not implemented in the camera.

---

### 5.5.2. Scalable methods

These methods control scalable parameters of the camera.  
Refer to instruction manual of the camera about scalable mode.

#### 5.5.2.1. GetSensorWidth method

This method gets number of active pixels in horizontal directions.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetSensorWidth(  
    out int sensorWidth  
)
```

[VB.NET]

```
Public Function GetSensorWidth (  
    <OutAttribute> ByRef sensorWidth As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSensorWidth(  
    [OutAttribute] Int32% sensorWidth  
)
```

##### [Parameters]

Parameters	Description
<i>sensorWidth</i>	A variable that receives number of active pixels in horizontal directions.

##### [Return value]

Returns result status.

---

### 5.5.2.2. GetSensorHeight method

This method gets number of active pixels in vertical directions.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetSensorHeight(  
    out int sensorHeight  
)
```

[VB.NET]

```
Public Function GetSensorHeight (  
    <OutAttribute> ByRef sensorHeight As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSensorHeight(  
    [OutAttribute] Int32% sensorHeight  
)
```

#### [Parameters]

Parameters	Description
<i>sensorHeight</i>	A variable that receives number of active pixels in vertical directions.

#### [Return value]

Returns result status.

---

### 5.5.2.3. GetRoi method

This method gets ROI (Region Of Interest) of the camera

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetRoi(  
    out int width,  
    out int height,  
    out int offsetX,  
    out int offsetY  
)
```

[VB.NET]

```
Public Function GetRoi (  
    <OutAttribute> ByRef width As Integer,  
    <OutAttribute> ByRef height As Integer,  
    <OutAttribute> ByRef offsetX As Integer,  
    <OutAttribute> ByRef offsetY As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetRoi(  
    [OutAttribute] Int32% width,  
    [OutAttribute] Int32% height,  
    [OutAttribute] Int32% offsetX,  
    [OutAttribute] Int32% offsetY  
)
```

#### [Parameters]

---

Parameters	Description
<i>width</i>	A variable that receives current width of the image in pixels.
<i>height</i>	A variable that receives current height of the image in pixels.
<i>offsetX</i>	A variable that receives current horizontal offset from the origin to the region of interest in pixels.
<i>offsetY</i>	A variable that receives current vertical offset from the origin to the region of interest in pixels.

#### [Return value]

---

Returns result status.

---

#### 5.5.2.4. SetRoi method

This method sets ROI (Region Of Interest) of the camera.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus SetRoi(  
    int width,  
    int height,  
    int offsetX,  
    int offsetY  
)
```

[VB.NET]

```
Public Function SetRoi (  
    width As Integer,  
    height As Integer,  
    offsetX As Integer,  
    offsetY As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetRoi(  
    Int32 width,  
    Int32 height,  
    Int32 offsetX,  
    Int32 offsetY  
)
```

##### [Parameters]

Parameters	Description
<i>width</i>	New width of the image in pixels.
<i>height</i>	New height of the image in pixels.
<i>offsetX</i>	New horizontal offset from the origin to the region of interest in pixels.
<i>offsetY</i>	New vertical offset from the origin to the region of interest in pixels.

##### [Return value]

Returns result status.

##### [Remarks]

ROI registers are protected from writing value during streaming is active.

---

### 5.5.2.5. GetWidthMinMax method

This method gets the maximum and minimum available image width and increment value of the image width.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetWidthMinMax(  
    out int widthMin,  
    out int widthMax,  
    out int widthInc  
)
```

[VB.NET]

```
Public Function GetWidthMinMax (  
    <OutAttribute> ByRef widthMin As Integer,  
    <OutAttribute> ByRef widthMax As Integer,  
    <OutAttribute> ByRef widthInc As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetWidthMinMax(  
    [OutAttribute] Int32% widthMin,  
    [OutAttribute] Int32% widthMax,  
    [OutAttribute] Int32% widthInc  
)
```

#### [Parameters]

---

Parameters	Description
<i>widthMin</i>	A variable that receives minimum available width.
<i>widthMax</i>	A variable that receives maximum available width.
<i>widthInc</i>	A variable that receives increment value for width parameter.

#### [Return value]

---

Returns result status.

---

### 5.5.2.6. GetWidth method

This method gets current image width in pixels.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetWidth(  
    out int width  
)
```

[VB.NET]

```
Public Function GetWidth (  
    <OutAttribute> ByRef width As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetWidth(  
    [OutAttribute] Int32% width  
)
```

#### [Parameters]

---

Parameters	Description
<i>width</i>	A variable that receives current width of the image in pixels.

#### [Return value]

---

Returns result status.



---

### 5.5.2.7. SetWidth method

This method sets image width in pixels.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetWidth(  
    int width  
)
```

[VB.NET]

```
Public Function SetWidth (  
    width As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetWidth(  
    Int32 width  
)
```

#### [Parameters]

Parameters	Description
<i>width</i>	New width of the image in pixels.

#### [Return value]

Returns result status.

#### [Remarks]

Image width register is protected from writing value during streaming is active.

---

### 5.5.2.8. GetHeightMinMax method

This method gets the maximum and minimum available image height and increment value of the image height.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetHeightMinMax(  
    out int heightMin,  
    out int heightMax,  
    out int heightInc  
)
```

[VB.NET]

```
Public Function GetHeightMinMax (  
    <OutAttribute> ByRef heightMin As Integer,  
    <OutAttribute> ByRef heightMax As Integer,  
    <OutAttribute> ByRef heightInc As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetHeightMinMax(  
    [OutAttribute] Int32% heightMin,  
    [OutAttribute] Int32% heightMax,  
    [OutAttribute] Int32% heightInc  
)
```

#### [Parameters]

---

Parameters	Description
<i>heightMin</i>	A variable that receives minimum available height.
<i>heightMax</i>	A variable that receives maximum available height.
<i>heightInc</i>	A variable that receives increment value for height parameter.

#### [Return value]

---

Returns result status.

---

### 5.5.2.9. GetHeight method

This method gets current image height in pixels.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetHeight(  
    out int height  
)
```

[VB.NET]

```
Public Function GetHeight (  
    <OutAttribute> ByRef height As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetHeight(  
    [OutAttribute] Int32% height  
)
```

#### [Parameters]

---

Parameters	Description
<i>height</i>	A variable that receives current height of the image in pixels.

#### [Return value]

---

Returns result status.

---

### 5.5.2.10. SetHeight method

This method sets image height in pixels.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetHeight(  
    int height  
)
```

[VB.NET]

```
Public Function SetHeight (  
    height As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetHeight(  
    Int32 height  
)
```

#### [Parameters]

Parameters	Description
<i>height</i>	New height of the image in pixels.

#### [Return value]

Returns result status.

#### [Remarks]

Image height register is protected from writing value during streaming is active.

---

### 5.5.2.11. GetOffsetXMinMax method

This method gets the maximum and minimum available horizontal image offset and increment value of image offset.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetOffsetXMinMax(  
    out int offsetXMin,  
    out int offsetXMax,  
    out int offsetXInc  
)
```

[VB.NET]

```
Public Function GetOffsetXMinMax (  
    <OutAttribute> ByRef offsetXMin As Integer,  
    <OutAttribute> ByRef offsetXMax As Integer,  
    <OutAttribute> ByRef offsetXInc As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetOffsetXMinMax(  
    [OutAttribute] Int32% offsetXMin,  
    [OutAttribute] Int32% offsetXMax,  
    [OutAttribute] Int32% offsetXInc  
)
```

#### [Parameters]

---

Parameters	Description
<i>offsetXMin</i>	A variable that receives minimum available offsetX.
<i>offsetXMax</i>	A variable that receives maximum available offsetX.
<i>offsetXInc</i>	A variable that receives increment value for offsetX parameter.

#### [Return value]

---

Returns result status.

---

### 5.5.2.12. GetOffsetX method

This method gets current horizontal image offset in pixels.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetOffsetX(  
    out int offsetX  
)
```

[VB.NET]

```
Public Function GetOffsetX (  
    <OutAttribute> ByRef offsetX As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetOffsetX(  
    [OutAttribute] Int32% offsetX  
)
```

#### [Parameters]

Parameters	Description
<i>offsetX</i>	A variable that receives current offsetX of the image in pixels.

#### [Return value]

Returns result status.

---

### 5.5.2.13. SetOffsetX method

This method sets horizontal image offset in pixels.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetOffsetX(  
    int offsetX  
)
```

[VB.NET]

```
Public Function SetOffsetX (  
    offsetX As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetOffsetX(  
    Int32 offsetX  
)
```

#### [Parameters]

Parameters	Description
<i>offsetX</i>	New offsetX of the image in pixels.

#### [Return value]

Returns result status.

#### [Remarks]

Image horizontal image offset register is protected from writing value during streaming is active.

---

#### 5.5.2.14. GetOffsetYMinMax method

This method gets the maximum and minimum available vertical image offset and increment value of image offset.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public CamApiStatus GetOffsetYMinMax(  
    out int offsetYMin,  
    out int offsetYMax,  
    out int offsetYInc  
)
```

[VB.NET]

```
Public Function GetOffsetYMinMax (  
    <OutAttribute> ByRef offsetYMin As Integer,  
    <OutAttribute> ByRef offsetYMax As Integer,  
    <OutAttribute> ByRef offsetYInc As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetOffsetYMinMax(  
    [OutAttribute] Int32% offsetYMin,  
    [OutAttribute] Int32% offsetYMax,  
    [OutAttribute] Int32% offsetYInc  
)
```

##### [Parameters]

---

Parameters	Description
<i>offsetYMin</i>	A variable that receives minimum available offsetY.
<i>offsetYMax</i>	A variable that receives maximum available offsetY.
<i>offsetYInc</i>	A variable that receives increment value for offsetY parameter.

##### [Return value]

---

Returns result status.



---

### 5.5.2.15. GetOffsetY method

This method gets current vertical image offset in pixels.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetOffsetY(  
    out int offsetY  
)
```

[VB.NET]

```
Public Function GetOffsetY (  
    <OutAttribute> ByRef offsetY As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetOffsetY(  
    [OutAttribute] Int32% offsetY  
)
```

#### [Parameters]

---

Parameters	Description
<i>offsetY</i>	A variable that receives current offsetY of the image in pixels.

#### [Return value]

---

Returns result status.

---

### 5.5.2.16. SetOffsetY method

This method sets vertical image offset in pixels.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetOffsetY(  
    int offsetY  
)
```

[VB.NET]

```
Public Function SetOffsetY (  
    offsetY As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetOffsetY(  
    Int32 offsetY  
)
```

#### [Parameters]

Parameters	Description
<i>offsetY</i>	New offsetY of the image in pixels.

#### [Return value]

Returns result status.

#### [Remarks]

Image vertical image offset register is protected from writing value during streaming is active.

---

### 5.5.3. Binning methods

These methods control binning feature of the camera  
Refer to instruction manual of the camera about binning feature.

#### 5.5.3.1. GetBinningHorizontalMinMax method

This method gets the maximum and minimum available horizontal binning mode value.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CamApiStatus GetBinningHorizontalMinMax(  
    out int min,  
    out int max,  
)
```

[VB.NET]

```
Public Function GetBinningHorizontalMinMax (  
    <OutAttribute> ByRef min As Integer,  
    <OutAttribute> ByRef max As Integer,  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetBinningHorizontalMinMax(  
    [OutAttribute] Int32% min,  
    [OutAttribute] Int32% max,  
)
```

---

**[Parameters]**

Parameters	Description
<i>min</i>	A variable that receives minimum available value.
<i>max</i>	A variable that receives maximum available value.

---

**[Return value]**

Returns result status.

---

**[Remarks]**

This method will return error status if "BinningHorizontal" register or node is not implemented in the camera.

Conditions that can get and set the binning parameters depend on the camera.  
For details, refer to the operation manual of the camera.

---

### 5.5.3.2. GetBinningHorizontal method

This method gets current horizontal binning mode value of the camera

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetBinningHorizontal(  
    out int value  
)
```

[VB.NET]

```
Public Function GetBinningHorizontal (  
    <OutAttribute> ByRef value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetBinningHorizontal(  
    [OutAttribute] Int32% value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current horizontal binning mode vaue.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "BinningHorizontal" register or node is not implemented in the camera.

Conditions that can get and set the binning parameters depend on the camera.  
For details, refer to the operation manual of the camera.

---

### 5.5.3.3. SetBinningHorizontal method

This method sets horizontal binning mode value of the camera

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetBinningHorizontal(  
    int value  
)
```

[VB.NET]

```
Public Function SetBinningHorizontal (  
    value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetBinningHorizontal(  
    Int32 value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New horizontal binning mode value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "BinningHorizontal" register or node is not implemented in the camera.

Conditions that can get and set the binning parameters depend on the camera.  
For details, refer to the operation manual of the camera.

"BinningHorizontal" register is protected from writing value during streaming is active.

---

#### 5.5.3.4. GetBinningVerticalMinMax method

This method gets the maximum and minimum available vertical binning mode value.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetBinningVerticalMinMax(  
    out int min,  
    out int max,  
)
```

[VB.NET]

```
Public Function GetBinningVerticalMinMax (  
    <OutAttribute> ByRef min As Integer,  
    <OutAttribute> ByRef max As Integer,  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetBinningVerticalMinMax(  
    [OutAttribute] Int32% min,  
    [OutAttribute] Int32% max,  
)
```

##### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives minimum available value.
<i>max</i>	A variable that receives maximum available value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "BinningVertical" register or node is not implemented in the camera.

Conditions that can get and set the binning parameters depend on the camera.

For details, refer to the operation manual of the camera.

---

### 5.5.3.5. GetBinningVertical method

This method gets current vertical binning mode value of the camera

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetBinningVertical(  
    out int value  
)
```

[VB.NET]

```
Public Function GetBinningVertical (  
    <OutAttribute> ByRef value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetBinningVertical(  
    [OutAttribute] Int32% value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current vertical binning mode value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “BinningVertical” register or node is not implemented in the camera.

Conditions that can get and set the binning parameters depend on the camera.  
For details, refer to the operation manual of the camera.

---

### 5.5.3.6. SetBinningVertical method

This method sets vertical binning mode value of the camera

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetBinningVertical(  
    int value  
)
```

[VB.NET]

```
Public Function SetBinningVertical (  
    value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetBinningVertical(  
    Int32 value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New vertical binning mode value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “BinningVertical” register or node is not implemented in the camera.

Conditions that can get and set the binning parameters depend on the camera.  
For details, refer to the operation manual of the camera.

“BinningVertical register” is protected from writing value during streaming is active.



---

#### 5.5.4. Decimation methods

These methods control decimation feature of the camera.  
Refer to instruction manual of the camera about decimation feature.

##### 5.5.4.1. GetDecimationHorizontalMinMax method

This method gets the maximum and minimum available horizontal decimation mode value.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CamApiStatus GetDecimationHorizontalMinMax(  
    out int min,  
    out int max,  
)
```

[VB.NET]

```
Public Function GetDecimationHorizontalMinMax (  
    <OutAttribute> ByRef min As Integer,  
    <OutAttribute> ByRef max As Integer,  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetDecimationHorizontalMinMax(  
    [OutAttribute] Int32% min,  
    [OutAttribute] Int32% max,  
)
```

---

**[Parameters]**

Parameters	Description
<i>min</i>	A variable that receives minimum available value.
<i>max</i>	A variable that receives maximum available value.

---

**[Return value]**

Returns result status.

---

**[Remarks]**

This method will return error status if "DecimationHorizontal" register or node is not implemented in the camera.

Conditions that can get and set the decimation parameters depend on the camera.  
For details, refer to the operation manual of the camera.

---

#### 5.5.4.2. GetDecimationHorizontal method

This method gets current horizontal decimation mode value of the camera

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetDecimationHorizontal(  
    out int value  
)
```

[VB.NET]

```
Public Function GetDecimationHorizontal (  
    <OutAttribute> ByRef value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetDecimationHorizontal(  
    [OutAttribute] Int32% value  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current horizontal decimation mode value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "DecimationHorizontal" register or node is not implemented in the camera.

Conditions that can get and set the decimation parameters depend on the camera.

For details, refer to the operation manual of the camera.

---

#### 5.5.4.3. SetDecimationHorizontal method

This method sets horizontal decimation mode value of the camera

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus SetDecimationHorizontal(  
    int value  
)
```

[VB.NET]

```
Public Function SetDecimationHorizontal (  
    value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetDecimationHorizontal(  
    Int32 value  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	New horizontal decimation mode value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “DecimationHorizontal” register or node is not implemented in the camera.

Conditions that can get and set the decimation parameters depend on the camera.

For details, refer to the operation manual of the camera.

Note that the camera may return success when invalid parameter value is specified to this method. The camera may not output proper image when invalid parameter is set to “DecimationHorizontal” register.

Make sure to set valid parameter to this method, referring instruction manual of the camera being used.

“DecimationHorizontal” register is protected from writing value during streaming is active.

---

#### 5.5.4.4. GetDecimationVerticalMinMax method

This method gets the maximum and minimum available vertical decimation mode value.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetDecimationVerticalMinMax(  
    out int min,  
    out int max,  
)
```

[VB.NET]

```
Public Function GetDecimationVerticalMinMax (  
    <OutAttribute> ByRef min As Integer,  
    <OutAttribute> ByRef max As Integer,  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetDecimationVerticalMinMax(  
    [OutAttribute] Int32% min,  
    [OutAttribute] Int32% max,  
)
```

##### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives minimum available value.
<i>max</i>	A variable that receives maximum available value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "DecimationVertical" register or node is not implemented in the camera.

Conditions that can get and set the decimation parameters depend on the camera.

For details, refer to the operation manual of the camera.

---

#### 5.5.4.5. GetDecimationVertical method

This method gets current vertical decimation mode value of the camera

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetDecimationVertical(  
    out int value  
)
```

[VB.NET]

```
Public Function GetDecimationVertical (  
    <OutAttribute> ByRef value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetDecimationVertical(  
    [OutAttribute] Int32% value  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current vertical decimation mode value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “DecimationVertical” register or node is not implemented in the camera.

Conditions that can get and set the decimation parameters depend on the camera.

For details, refer to the operation manual of the camera.

---

#### 5.5.4.6. SetDecimationVertical method

This method sets vertical decimation mode value of the camera

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus SetDecimationVertical(  
    int value  
)
```

[VB.NET]

```
Public Function SetDecimationVertical (  
    value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus SetDecimationVertical(  
        Int32 value  
    )
```

##### [Parameters]

Parameters	Description
<i>value</i>	New vertical decimation mode value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “DecimationVertical” register or node is not implemented in the camera.

Conditions that can get and set the decimation parameters depend on the camera.

For details, refer to the operation manual of the camera.

Note that the camera may return success when invalid parameter value is specified to this method. The camera may not output proper image when invalid parameter is set to “DecimationHorizontal” register.

Make sure to set valid parameter to this method, referring instruction manual of the camera being used.

“DecimationVertical” register is protected from writing value during streaming is active.

---

### 5.5.5. Reverse methods

These methods control image reverse (mirroring) feature of the camera  
Refer to instruction manual of the camera about Reverse.

#### 5.5.5.1. GetReverseX method

This method gets current horizontal image reverse mode value of the camera.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetReverseX(  
    out bool value  
)
```

[VB.NET]

```
Public Function GetReverseX (  
    <OutAttribute> ByRef value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetReverseX(  
    [OutAttribute] Boolean% value  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current horizontal image reverse mode value. If true, horizontal image reverse feature is enabled.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "ReverseX" register or node is not implemented in the camera.

---

### 5.5.5.2. SetReverseX method

This method sets horizontal image reverse mode value of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetReverseX(  
    bool value  
)
```

[VB.NET]

```
Public Function SetReverseX (  
    value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetReverseX(  
    Boolean value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New horizontal image reverse mode value. If true, horizontal image reverse feature will be enabled.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "ReverseX" register or node is not implemented in the camera.

"ReverseX" register is protected from writing value during streaming is active.



---

### 5.5.5.3. GetReverseY method

This method gets current vertical image reverse mode value of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetReverseY(  
    out bool value  
)
```

[VB.NET]

```
Public Function GetReverseY (  
    <OutAttribute> ByRef value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetReverseY(  
    [OutAttribute] Boolean% value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current vertical image reverse mode value. If true, vertical image reverse feature is enabled.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "ReverseY" register or node is not implemented in the camera.

---

#### 5.5.5.4. SetReverseY method

This method sets vertical image reverse mode value of the camera.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus SetReverseY(  
    bool value  
)
```

[VB.NET]

```
Public Function SetReverseY (  
    value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetReverseY(  
    Boolean value  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	New vertical image reverse mode value. If true, vertical image reverse feature will be enabled.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “ReverseY” register or node is not implemented in the camera.

“ReverseY” register is protected from writing value during streaming is active.

---

### 5.5.6. PixelFormat methods

These methods control [PixelFormat](#) of image stream.

Refer to instruction manual of the camera or section 6.5 of document “How to Use GenApi with TeliGevSDK and TeliU3vSDK” downloadable from our home page, about [PixelFormat](#).

#### 5.5.6.1. GetPixelFormat method

This method gets [PixelFormat](#) of image stream.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetPixelFormat(  
    out CameraPixelFormat pixelFormat  
)
```

[VB.NET]

```
Public Function GetPixelFormat (  
    <OutAttribute> ByRef pixelFormat As CameraPixelFormat  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetPixelFormat(  
    [OutAttribute] CameraPixelFormat% pixelFormat  
)
```

##### [Parameters]

Parameters	Description
<i>pixelFormat</i>	A variable that receives current <a href="#">PixelFormat</a> of the image stream.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “PixelFormat” register or node is not implemented in the camera.

---

### 5.5.6.2. SetPixelFormat method

This method sets [PixelFormat](#) of image stream.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetPixelFormat(  
    CameraPixelFormat pixelFormat  
)
```

[VB.NET]

```
Public Function SetPixelFormat (  
    pixelFormat As CameraPixelFormat  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetPixelFormat(  
    CameraPixelFormat pixelFormat  
)
```

#### [Parameters]

Parameters	Description
<i>pixelFormat</i>	New <a href="#">PixelFormat</a> of the image stream.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “PixelFormat” register or node is not implemented in the camera.

In USB3 Vision camera case, this function will convert the specified “[PixelFormat](#)” value to “PixelCoding” and “PixelFormatSize” parameter values, and write them to corresponding registers.

In GigE Vision camera case, the specified “[PixelFormat](#)” value will be written to “PixelFormat” register, as is.

“PixelFormat” register or “PixelCoding” and “PixelFormatSize” registers are protected from writing value during streaming is active.

---

### 5.5.7. TestPattern methods

These method control the [test pattern](#) (test image) generationfunction of the camera.  
Refer to instruction manual of the camera about [test pattern](#) (test image).

#### 5.5.7.1. GetTestPattern method

This method gets current [test pattern](#) (test image) choice.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CamApiStatus GetTestPattern(  
    out CameraTestPattern testPattern  
)
```

[VB.NET]

```
Public Function GetTestPattern (  
    <OutAttribute> ByRef testPattern As CameraTestPattern  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTestPattern(  
    [OutAttribute] CameraTestPattern% testPattern  
)
```

---

**[Parameters]**

Parameters	Description
<i>testPattern</i>	A variable that receives current <a href="#">test pattern</a> choice.

---

**[Return value]**

Returns result status.

---

**[Remarks]**

This method will return error status if “TestPattern” or “TestImageSelector” register or node is not implemented in the camera.

---

### 5.5.7.2. SetTestPattern method

This method sets [test pattern](#) (test image) choice.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetTestPattern(  
    CameraTestPattern testPattern  
)
```

[VB.NET]

```
Public Function SetTestPattern (  
    testPattern As CameraTestPattern  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetTestPattern(  
    CameraTestPattern testPattern  
)
```

#### [Parameters]

Parameters	Description
<i>testPattern</i>	New <a href="#">test pattern</a> .

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “TestPattern” or “TestImageSelector” register or node is not implemented in the camera.

---

### 5.5.8. AcquisitionControl methods

These methods control image acquisition feature of the camera.  
Refer ti instructuin manual about image acquisition.

#### 5.5.8.1. GetStreamPayloadSize mesod

This method gets current payload size of image stream.  
Payload size is usually same as image size.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetStreamPayloadSize(  
    out int payloadSize  
)
```

[VB.NET]

```
Public Function GetStreamPayloadSize (  
    <OutAttribute> ByRef payloadSize As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetStreamPayloadSize(  
    [OutAttribute] Int32% payloadSize  
)
```

##### [Parameters]

Parameters	Description
<i>payloadSize</i>	A variable that receives current payload size (image size) of the image stream in bytes.

##### [Return value]

Returns result status.

---

### 5.5.8.2. GetStreamEnable method

This method gets streaming status (enabled / disabled) of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetStreamEnable(  
    out bool value  
)
```

[VB.NET]

```
Public Function GetStreamEnable (  
    <OutAttribute> ByRef value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetStreamEnable(  
    [OutAttribute] Boolean % value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current status if image streaming. If true, image streaming is active.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for USB3 Vision camera.



---

### 5.5.8.3. GetAcquisitionFrameCountMinMax method

This method gets the maximum and minimum available “AcquisitionFrameCount” value.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public CamApiStatus GetAcquisitionFrameCountMinMax(  
    out int min,  
    out int max  
)
```

[VB.NET]

```
Public Function GetAcquisitionFrameCountMinMax (  
    <OutAttribute> ByRef min As Integer,  
    <OutAttribute> ByRef max As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetAcquisitionFrameCountMinMax(  
    [OutAttribute] Int32% min,  
    [OutAttribute] Int32% max  
)
```

#### **[Parameters]**

---

Parameters	Description
<i>min</i>	A variable that receives the minimum available AcquisitionFrameCount value.
<i>max</i>	A variable that receives the minimum available AcquisitionFrameCount value.

#### **[Return value]**

---

Returns result status.

#### **[Remarks]**

---

This method will return error status if “AcquisitionFrameCount” register or node is not implemented in the camera.

---

#### 5.5.8.4. GetAcquisitionFrameCount method

This method gets current “AcquisitionFrameCount” value.

In MultiFrame acquisition mode, the camera will stop image acquisition when acquired image count reached to “AcquisitionFrameCount” value.

In ImageBufferRead mode, the camera will send “AcquisitionFrameCount” value frames of images from image buffer in the camera for an “[ExecuteImageBufferRead\(\)](#)” method.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetAcquisitionFrameCount(  
    out int frameCount  
)
```

[VB.NET]

```
Public Function GetAcquisitionFrameCount (  
    <OutAttribute> ByRef frameCount As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetAcquisitionFrameCount(  
    [OutAttribute] Int32% frameCount  
)
```

##### [Parameters]

Parameters	Description
<i>frameCount</i>	A variable that receives current AcquisitionFrameCount value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “AcquisitionFrameCount” register or node is not implemented in the camera.

---

#### 5.5.8.5. SetAcquisitionFrameCount method

This method sets "AcquisitionFrameCount" value.

In MultiFrame acquisition mode, the camera will stop image acquisition when acquired image count reached to "AcquisitionFrameCount" value.

In ImageBufferRead mode, the camera will send "AcquisitionFrameCount" value frames of images from image buffer in the camera for an "[ExecuteImageBufferRead\(\)](#)" method.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus SetAcquisitionFrameCount(  
    int frameCount  
)
```

[VB.NET]

```
Public Function SetAcquisitionFrameCount (  
    frameCount As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetAcquisitionFrameCount(  
    Int32 frameCount  
)
```

##### [Parameters]

Parameters	Description
<i>frameCount</i>	New AcquisitionFrameCount value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "AcquisitionFrameCount" register or node is not implemented in the camera.

"AcquisitionFrameCount" registers is protected from writing value during streaming is active.

---

### 5.5.8.6. GetAcquisitionFrameRateControl method

This method gets current "AcquisitionFrameRateControl" register value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetAcquisitionFrameRateControl(  
    out CameraAcqFrameRateCtrl value  
)
```

[VB.NET]

```
Public Function GetAcquisitionFrameRateControl (  
    <OutAttribute> ByRef value As CameraAcqFrameRateCtrl  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetAcquisitionFrameRateControl(  
    [OutAttribute] CameraAcqFrameRateCtrl% value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current <a href="#">AcquisitionFrameRateControl</a> value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "AcquisitionFrameRateControl" register or node is not implemented in the camera.

---

### 5.5.8.7. SetAcquisitionFrameRateControl method

This method sets “AcquisitionFrameRateControl” register value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetAcquisitionFrameRateControl(  
    CameraAcqFrameRateCtrl value  
)
```

[VB.NET]

```
Public Function SetAcquisitionFrameRateControl (  
    value As CameraAcqFrameRateCtrl  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetAcquisitionFrameRateControl(  
    CameraAcqFrameRateCtrl value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New <a href="#">AcquisitionFrameRateControl</a> value.

#### [Return value]

Returns result status.

#### [Remarks]

If register value is “CameraAcqFrameRateCtrl.NoSpecify”, frame rate will be decided using parameters other than “AcquisitionFrameRate” register and the decided frame rate value will be read out from “AcquisitionFrameRate” register.

If register value is “CameraAcqFrameRateCtrl.Manual”, frame rate will be controlled to register value of “AcquisitionFrameRate” register.

This method will return error status if “AcquisitionFrameRateControl” register or node is not implemented in the camera.

---

#### 5.5.8.8. GetAcquisitionFrameRateMinMax method

This method gets the maximum and minimum available frame rate value

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetAcquisitionFrameRateMinMax(  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetAcquisitionFrameRateMinMax (  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetAcquisitionFrameRateMinMax(  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

##### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives the minimum available AcquisitionFrameRate value.
<i>max</i>	A variable that receives the minimum available AcquisitionFrameRate value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "AcquisitionFrameRate" register or node is not implemented in the camera.

---

#### 5.5.8.9. GetAcquisitionFrameRate method

This method gets current frame rate value.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetAcquisitionFrameRate(  
    out double frameRate  
)
```

[VB.NET]

```
Public Function GetAcquisitionFrameRate (  
    <OutAttribute> ByRef frameRate As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetAcquisitionFrameRate(  
    [OutAttribute] Double% frameRate  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current AcquisitionFrameRate value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "AcquisitionFrameRate" register or node is not implemented in the camera.

---

#### 5.5.8.10. SetAcquisitionFrameRate method

This method sets frame rate value.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus SetAcquisitionFrameRate(  
    double frameRate  
)
```

[VB.NET]

```
Public Function SetAcquisitionFrameRate (  
    frameRate As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetAcquisitionFrameRate(  
    double frameRate  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	New AcquisitionFrameRate value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "AcquisitionFrameRate" register or node is not implemented in the camera.

Availability of changing "AcquisitionFrameRate" register value during streaming is active depends on model of the camera.



---

### 5.5.9. ImageBuffer methods

These methods controls feature of image buffer in the camera.  
Refer to instruction manual of the camera about image buffer feature

#### 5.5.9.1. GetImageBufferMode method

This method gets current "[ImageBufferMode](#)" status of the camera.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CamApiStatus GetImageBufferMode(  
    out CameraImageBufferMode mode  
)
```

[VB.NET]

```
Public Function GetImageBufferMode (  
    <OutAttribute> ByRef mode As CameraImageBufferMode  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetImageBufferMode(  
    [OutAttribute] CameraImageBufferMode% mode  
)
```

---

**[Parameters]**

Parameters	Description
<i>mode</i>	A variable that receives current <a href="#">ImageBufferMode</a> status.

---

**[Return value]**

Returns result status.

---

**[Remarks]**

This method will return error status if "ImageBufferMode" register or node is not implemented in the camera.

---

### 5.5.9.2. SetImageBufferMode mode

This method sets "[ImageBufferMode](#)" status (on /off) of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus SetImageBufferMode(  
    CameraImageBufferMode mode  
)
```

[VB.NET]

```
Public Function SetImageBufferMode (  
    mode As CameraImageBufferMode  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus SetImageBufferMode(  
        CameraImageBufferMode mode  
)
```

#### [Parameters]

---

Parameters	Description
<i>mode</i>	New <a href="#">ImageBufferMode</a> status (on / off).

#### [Return value]

---

Returns result status.

#### [Remarks]

---

This method will return error status if "ImageBufferMode" register or node is not implemented in the camera.

"ImageBufferMode" register is protected from writing value during streaming is active.

---

### 5.5.9.3. GetImageBufferFrameCount method

This method gets number of images stored in image buffer of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetImageBufferFrameCount(  
    out int frameCount  
)
```

[VB.NET]

```
Public Function GetImageBufferFrameCount (  
    <OutAttribute> ByRef frameCount As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetImageBufferFrameCount(  
    [OutAttribute] Int32% frameCount  
)
```

#### [Parameters]

Parameters	Description
<i>frameCount</i>	A variable that receives number of images stored in camera image buffer.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "ImageBufferMode" register or node is not implemented in the camera.

---

#### 5.5.9.4. ExecuteImageBufferRead method

This method sends command for reading out images from image buffer of the camera.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus ExecuteImageBufferRead()
```

[VB.NET]

```
Public Function ExecuteImageBufferRead As CamApiStatus
```

[C++]

```
public:  
CamApiStatus ExecuteImageBufferRead()
```

##### [Return value]

Returns result status.

##### [Remarks]

Frame count sent from the camera for a single "[ExecuteImageBufferRead\(\)](#)" method call can be set using "[SetAcquisitionFrameCount\(\)](#)" method.

Use methods of CameraStream object for receiving images read out from image buffer of the camera.

If not enough images were stored in image buffer of the camera when "[ExecuteImageBufferRead\(\)](#)" was called, the rest images will be transferred to the ImageRingBuffer as soon as they are saved to the image buffer of the camera.

This method will return error status if "ImageBufferMode" register or node is not implemented in the camera.

---

## 5.5.10.TriggerControl methods

These methods control trigger function of the camea

### 5.5.10.1. GetTriggerMode method

This method gets current triggering mode status.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetTriggerMode(  
    out bool value  
)
```

[VB.NET]

```
Public Function GetTriggerMode (  
    <OutAttribute> ByRef value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTriggerMode(  
    [OutAttribute] Boolean% value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current TriggerMode status. If true, TriggerMode is On.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TriggerMode" register or node is not implemented in the camera.

---

### 5.5.10.2. SetTriggerMode method

This method sets triggering mode status.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetTriggerMode(  
    bool value  
)
```

[VB.NET]

```
Public Function SetTriggerMode (  
    value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetTriggerMode(  
    Boolean value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New TriggerMode status. If true, TriggerMode is On.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TriggerMode" register or node is not implemented in the camera.

Availability of changing "TriggerMode" register value during streaming is active depends on model of the camera.

---

### 5.5.10.3. GetTriggerSequence method

This method gets current [sequence of exposure time control](#).

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetTriggerSequence(  
    out CameraTriggerSequence sequence  
)
```

[VB.NET]

```
Public Function GetTriggerSequence (  
    <OutAttribute> ByRef sequence As CameraTriggerSequence  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTriggerSequence(  
    [OutAttribute] CameraTriggerSequence% sequence  
)
```

#### [Parameters]

Parameters	Description
<i>sequence</i>	A variable that receives current <a href="#">sequence of exposure time control</a> .

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “TriggerSequence” or “ExposureMode” / “TriggerSelector” register (or node) is not implemented in the camera.

Refer to instruction manual of the camera or section 2 of document “TriggerSequence” downloadable from [our web site](#), about trigger sequence.

---

#### 5.5.10.4. SetTriggerSequence method

This method sets [sequence of exposure time control](#).

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus SetTriggerSequence(  
    CameraTriggerSequence sequence  
)
```

[VB.NET]

```
Public Function SetTriggerSequence (  
    sequence As CameraTriggerSequence  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetTriggerSequence(  
    CameraTriggerSequence sequence  
)
```

##### [Parameters]

Parameters	Description
<i>sequence</i>	New <a href="#">sequence of exposure time control</a> .

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “TriggerSequence” or “ExposureMode” / “TriggerSelector” register (or node) is not implemented in the camera.

Refer to instruction manual of the camera or section 2 of document “TriggerSequence” downloadable from [our web site](#), about trigger sequence.



---

### 5.5.10.5. GetTriggerSource method

This method gets current trigger source signal for random trigger shutter mode.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetTriggerSource(  
    out CameraTriggerSource source  
)
```

[VB.NET]

```
Public Function GetTriggerSource (  
    <OutAttribute> ByRef source As CameraTriggerSource  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTriggerSource(  
    [OutAttribute] CameraTriggerSource% source  
)
```

#### [Parameters]

Parameters	Description
<i>source</i>	A variable that receives current <a href="#">trigger source</a>

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TriggerSource" register or node is not implemented in the camera.

---

### 5.5.10.6. SetTriggerSource method

This method sets trigger source signal for random trigger shutter mode.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetTriggerSource(  
    CameraTriggerSource source  
)
```

[VB.NET]

```
Public Function SetTriggerSource (  
    source As CameraTriggerSource  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetTriggerSource(  
    CameraTriggerSource source  
)
```

#### [Parameters]

Parameters	Description
<i>source</i>	New <a href="#">trigger source</a>

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TriggerSource" register or node is not implemented in the camera.

---

### 5.5.10.7. GetTriggerAdditionalParameterMinMax method

This method gets the maximum and minimum available value for “TriggerAdditionalParameter” or “AcquisitionBurstFrameCount”.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetTriggerAdditionalParameterMinMax(  
    out int min,  
    out int max  
)
```

[VB.NET]

```
Public Function GetTriggerAdditionalParameterMinMax (  
    <OutAttribute> ByRef min As Integer,  
    <OutAttribute> ByRef max As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTriggerAdditionalParameterMinMax(  
    [OutAttribute] Int32% min,  
    [OutAttribute] Int32% max  
)
```

#### [Parameters]

---

Parameters	Description
<i>min</i>	A variable that receives the minimum available TriggerAdditionalParameter or AcquisitionBurstFrameCount value.
<i>max</i>	A variable that receives the minimum available TriggerAdditionalParameter or AcquisitionBurstFrameCount value.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

In BU series camera, TriggerAdditionalParameter register is used as “number of frames acquired by a single trigger signal” in Bulk mode.

In BG series camera, AcquisitionBurstFrameCount register is used as “number of frames acquired by a single trigger signal” in FrameBurst mode.

This method will return error status if those registers or nodes are not implemented in the camera.

---

#### 5.5.10.8. GetTriggerAdditionalParameter method

This method gets current “TriggerAdditionalParameter” or “AcquisitionBurstFrameCount” value.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetTriggerAdditionalParameter(  
    out int value  
)
```

[VB.NET]

```
Public Function GetTriggerAdditionalParameter (  
    <OutAttribute> ByRef value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTriggerAdditionalParameter(  
    [OutAttribute] Int32% value  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current TriggerAdditionalParameter or AcquisitionBurstFrameCount value.

##### [Return value]

Returns result status.

##### [Remarks]

In BU series camera, TriggerAdditionalParameter register is used as “number of frames acquired by a single trigger signal” in Bulk mode.

In BG series camera, AcquisitionBurstFrameCount register is used as “number of frames acquired by a single trigger signal” in FrameBurst mode.

This method will return error status if those registers or nodes are not implemented in the camera.

---

### 5.5.10.9. SetTriggerAdditionalParameter method

This method sets “TriggerAdditionalParameter” or “AcquisitionBurstFrameCount” value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetTriggerAdditionalParameter(  
    int value  
)
```

[VB.NET]

```
Public Function SetTriggerAdditionalParameter (  
    value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetTriggerAdditionalParameter(  
    Int32 value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New TriggerAdditionalParameter or AcquisitionBurstFrameCount value.

#### [Return value]

Returns result status.

#### [Remarks]

In BU series camera, TriggerAdditionalParameter register is used as “number of frames acquired by a single trigger signal” in Bulk mode.

In BG series camera, AcquisitionBurstFrameCount register is used as “number of frames acquired by a single trigger signal” in FrameBurst mode.

This method will return error status if those registers or nodes are not implemented in the camera.

---

#### 5.5.10.10. GetTriggerDelayMinMax method

This method gets the maximum and minimum value available for “TriggerDeley” register.

“TriggerDelay” register controls delay time from detection of trigger signal until starting exposure.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public CamApiStatus GetTriggerDelayMinMax(  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetTriggerDelayMinMax (  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTriggerDelayMinMax(  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

##### [Parameters]

---

Parameters	Description
<i>min</i>	A variable that receives the minimum available TriggerDelay value in microseconds.
<i>max</i>	A variable that receives the minimum available TriggerDeay value in microseconds.

##### [Return value]

---

Returns result status.

##### [Remarks]

---

This method will return error status if “TriggerDelay” register or node is not implemented in the camera.

---

### 5.5.10.11. GetTriggerDelay method

This method gets current delay time from detection of trigger signal until starting exposure.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetTriggerDelay(  
    out double microseconds  
)
```

[VB.NET]

```
Public Function GetTriggerDelay (  
    <OutAttribute> ByRef microseconds As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTriggerDelay(  
    [OutAttribute] Double% microseconds  
)
```

#### [Parameters]

Parameters	Description
<i>microseconds</i>	A variable that receives current TriggerDelay value in microseconds.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TriggerDelay" register or node is not implemented in the camera.

---

### 5.5.10.12. SetTriggerDelay method

This method gets current delay time from detection of trigger signal until starting exposure.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetTriggerDelay(  
    double microseconds  
)
```

[VB.NET]

```
Public Function SetTriggerDelay (  
    microseconds As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetTriggerDelay(  
    Double microseconds  
)
```

#### [Parameters]

Parameters	Description
<i>microseconds</i>	A variable that receives current TriggerDelay value in microseconds.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TriggerDelay" register or node is not implemented in the camera.



---

### 5.5.10.13. ExecuteSoftwareTrigger method

This method sends software trigger command to the camera.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public CamApiStatus ExecuteSoftwareTrigger()
```

[VB.NET]

```
Public Function ExecuteSoftwareTrigger As CamApiStatus
```

[C++]

```
public:  
CamApiStatus ExecuteSoftwareTrigger()
```

#### **[Return value]**

---

Returns result status.

#### **[Remarks]**

---

If this method is executed during the camera is not ready for accepting software trigger command, the camera may return success to this method with doing nothing.

---

### 5.5.11.ExposureTime methods

These methods control exposure of the camera.

Refer to instruction manual of the camera about exposure control.

#### 5.5.11.1. GetExposureTimeControl method

This method gets current exposure time control mode of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public CamApiStatus GetExposureTimeControl(  
    out
```

---

```
CameraExposureTimeCtrl value  
)
```

```
[VB.NET]
```

```
Public Function GetExposureTimeControl (  
    <OutAttribute> ByRef value As
```

---

CameraExposureTimeCtrl

) As [CamApiStatus](#)

[C++]

public:

[CamApiStatus](#) GetExposureTimeControl(

[OutAttribute]

---

CameraExposureTimeCtrl% *value*  
)

**[Parameters]**

Parameters	Description
<i>value</i>	A variable that receives current <a href="#">ExposureTimeControl</a> value.

**[Return value]**

Returns result status.

**[Remarks]**

Refer to [7.19](#)

---

CameraExposureTimeCtrl about control state.

For example, when “TriggerMode” is On and “TriggerSequence” is TriggerSequence1, exposure time will be controlled by trigger pulse width. In USB3 Vision camera, “ExposureTimeControl1” will be

---

CameraExposureTimeCtrl.NoSpecify.

In USB3 Vision camera case, this method writes “ExposureTimeControl” register value to [\*value\*](#) parameter, as is.

In GigE Vision camera case, this method creates a [“CameraExposureTimeCtrl” type value](#) corresponding to “ExposureAuto” register value and writes it to [\*value\*](#) parameter.

This method will return error status if “ExposureTimeControl” or “ExposureAuto” register (or node) is not implemented in the camera.

---

#### 5.5.11.2. SetExposureTimeControl method

This method sets [exposure time control mode](#).

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public CamApiStatus SetExposureTimeControl(  
    out
```



---

```
CameraExposureTimeCtrl value
)
```

```
[VB.NET]
```

```
Public Function SetExposureTimeControl (
    value As
```

---

CameraExposureTimeCtrl

) As [CamApiStatus](#)

[C++]

public:

[CamApiStatus](#) SetExposureTimeControl(

[OutAttribute]

---

CameraExposureTimeCtrl *value*  
)

**[Parameters]**

Parameters	Description
<i>value</i>	New CameraExposureTimeCtrl value.

**[Return value]**

Returns result status.

**[Remarks]**

Refer to [7.19](#)

---

CameraExposureTimeCtrl about control state.

For example, when “TriggerMode” is On and “TriggerSequence” is TriggerSequence1, exposure time will be controlled by trigger pulse width. In USB3 Vision camera, “ExposureTimeControl1” will be

---

CameraExposureTimeCtrl.NoSpecify.

In USB3 Vision camera case, this method writes [value](#) parameter to “ExposureTimeControl” register, as is.

In GigE Vision camera case, this method creates a value corresponding to [value](#) parameter and writes it to “ExposureAuto” register.

This method will return error status if “ExposureTimeControl” or “ExposureAuto” register (or node) is not implemented in the camera.

---

### 5.5.11.3. GetExposureTimeMinMax method

This method gets the maximum and minimum available exposure time value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetExposureTimeMinMax(  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetExposureTimeMinMax (  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetExposureTimeMinMax(  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

#### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives the minimum available ExposureTime value in microseconds.
<i>max</i>	A variable that receives the minimum available ExposureTime value in microseconds.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “ExposureTime” register or node is not implemented in the camera.

---

#### 5.5.11.4. GetExposureTime method

This method gets current exposure time.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetExposureTime(  
    out double microseconds  
)
```

[VB.NET]

```
Public Function GetExposureTime (  
    <OutAttribute> ByRef microseconds As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetExposureTime(  
    [OutAttribute] Double% microseconds  
)
```

##### [Parameters]

Parameters	Description
<i>microseconds</i>	A variable that receives current "ExposureTime" value in microseconds.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "ExposureTime" register or node is not implemented in the camera.

---

#### 5.5.11.5. SetExposureTime method

This method sets exposure time to the camera, which will be used when “ExposureTimeControl” register value is “



---

CameraExposureTimeCtrl.Manual”.

namespace: Teli.TeliCamAPI.NET

**[Syntax]**

---

[C#]

```
public CamApiStatus SetExposureTime(  
    double microseconds  
)
```

[VB.NET]

```
Public Function SetExposureTime (  
    microseconds As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetExposureTime(  
    Double microseconds  
)
```

**[Parameters]**

---

Parameters	Description
<i>microseconds</i>	A variable that receives current “ExposureTime” value in microseconds.

**[Return value]**

---

Returns result status.

**[Remarks]**

---

This method will return error status if “ExposureTime” register or node is not implemented in the camera.

---

## 5.5.12.DigitalI/OControl methods

These methods control digital I/O ports of the camera.  
Refer to instruction manual of the camera about digital I/O port.

### 5.5.12.1. GetLineModeAll method

This method gets information of all I/O lines in the camera whether the line is input line or output line.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetLineModeAll(  
    out uint value  
)
```

[VB.NET]

```
Public Function GetLineModeAll (  
    <OutAttribute> ByRef value As UInteger  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetLineModeAll(  
    [OutAttribute] UInt32% value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current information of all I/O lines in the camera.  Each bit corresponds to each I/O line.(Line0 : Bit0 , Line1 : Bit1 , Line2 : Bit2 , . . .)  If bit value is 0, the line is input line, If bit value is 1, the line is output line. For example, If <i>value</i> is 0x06, Line0 : input, Line1 : output, Line2 : output , . . .)

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “LineModeAll” register, or “LineSelector” and “LineMode” registers, (or nodes) are not implemented in the camera.

In USB3 Vision camera case, this method will return “LineModeAll” register value as is.

In GigE Vision camera case, this method will create *value* parameter data using “[LineSelector](#)” and “LineMode” registers.

Refer to instruction manual of the camera about registers mentioned.

---

### 5.5.12.2. SetLineModeAll method

This method sets information of all I/O lines in the camera whether the line is input line or output line.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

[C#]

```
public CamApiStatus SetLineModeAll(  
    uint value  
)
```

[VB.NET]

```
Public Function SetLineModeAll (  
    value As UInteger  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetLineModeAll(  
    UInt32 value  
)
```

#### **[Parameters]**

Parameters	Description
<i>value</i>	New information of all I/O lines in the camera whether the line is input line or output line  Each bit corresponds to each I/O line.(Line0 : Bit0 , Line1 : Bit1 , Line2 : Bit2 , . .)  If bit value is 0, the line is input line, If bit value is 1, the line is output line. For example, If <i>puiValue</i> is 0x06, Line0 : input, Line1 : output, Line2 : output , .)

#### **[Return value]**

Returns result status.

#### **[Remarks]**

This method will return error status if “LineModeAll” register, or “LineSelector” and “LineMode” registers, (or nodes) are not implemented in the camera.

In USB3 Vision camera case, this method will return “LineModeAll” register value as is.

In GigE Vision camera case, this method will create *value* parameter data using “[LineSelector](#)” and “LineMode” registers.

Refer to instruction manual of the camera about registers mentioned.

---

### 5.5.12.3. GetLineInverterAll method

This method gets current polarity of all I/O lines in the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetLineInverterAll(  
    out uint value  
)
```

[VB.NET]

```
Public Function GetLineInverterAll (  
    <OutAttribute> ByRef value As UInteger  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetLineInverterAll(  
    [OutAttribute] UInt32% value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	<p>A variable that receives current polarity of all I/O lines in the camera.</p> <p>Each bit corresponds to each I/O line. (Line0 : Bit0 , Line1 : Bit1 , Line2 : Bit2 , . . .)</p> <p>If bit value is 0, the line is not inverted, If bit value is 1, the line is inverted.</p> <p>For example, If <i>value</i> is 0x06, Line0 : not inverted, Line1 : inverted, Line2 : inverted , . . .</p>

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “LineInverterAll” register, or “LineSelector” and “LineInverter” registers, (or nodes) are not implemented in the camera.

In USB3 Vision camera case, this method will write “LineInverterAll” register value to *value* parameter, as is.

In GigE Vision camera case, this method will create *value* parameter data using “[LineSelector](#)” and “LineInverter” registers.

Refer to instruction manual about registers mentioned.

---

#### 5.5.12.4. SetLineInverterAll method

This method sets polarity of all I/O lines in the camera.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus SetLineInverterAll(  
    uint value  
)
```

[VB.NET]

```
Public Function SetLineInverterAll (  
    value As UInteger  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetLineInverterAll(  
    UInt32 value  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	<p>New polarity of all I/O lines in the camera.</p> <p>Each bit corresponds to each I/O line. (Line0 : Bit0 , Line1 : Bit1 , Line2 : Bit2 , . . . )</p> <p>If bit value is 0, the line is not inverted, If bit value is 1, the line is inverted.</p> <p>For example, If <i>puiValue</i> is 0x06, Line0 : not inverted, Line1 : inverted, Line2 : inverted , . . .</p>

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “LineInverterAll” register, or “LineSelector” and “LineInverter” registers, (or nodes) are not implemented in the camera.

In USB3 Vision camera case, this method will write *value* parameter to “LineInverterAll” register, as is. In GigE Vision camera case, this method will write *value* parameters using “[LineSelector](#)” and “LineInverter” registers.

This method may return error status if bit value 1 ( invert) is set to ports that does not accept invert feature. For example, writing 1 to Line0 is not allowed in GigE Vision camera.

Refer to instruction manual of the camera about registers mentioned.

---

### 5.5.12.5. GetLineStatusAll method

This method gets current status of all I/O lines in the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetLineStatusAll(  
    out uint value  
)
```

[VB.NET]

```
Public Function GetLineStatusAll (  
    <OutAttribute> ByRef value As UInteger  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetLineStatusAll(  
    [OutAttribute] UInt32% value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current status of all I/O lines in the camera.  Each bit corresponds to each I/O line. (Line0 : Bit0 , Line1 : Bit1 , Line2 : Bit2 , . )

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "LineStatusAll" register or node is not implemented in the camera.

---

### 5.5.12.6. GetUserOutputValueAll method

This method gets current output value of all user output lines in the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetUserOutputValueAll(  
    out uint value  
)
```

[VB.NET]

```
Public Function GetUserOutputValueAll (  
    <OutAttribute> ByRef value As UInteger  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetUserOutputValueAll(  
    [OutAttribute] Int32% value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current output value of all user output lines in the camera.  Each bit corresponds to each I/O line. (Line0 : Bit0 , Line1 : Bit1 , Line2 : Bit2 , . )

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "LineOutputValueAll" register or node is not implemented in the camera.

---

### 5.5.12.7. SetUserOutputValueAll method

This method sets output value of all user output lines in the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetUserOutputValueAll(  
    uint value  
)
```

[VB.NET]

```
Public Function SetUserOutputValueAll (  
    value As UInteger  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetUserOutputValueAll(  
    UInt32 value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New output value of all user output lines in the camera.  Each bit corresponds to each I/O line. (Line0 : Bit0 , Line1 : Bit1 , Line2 : Bit2 , . . )

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “LineOutputValueAll” register or node is not implemented in the camera.

This method ignores bit values whose corresponding I/O line is not UserOutput.



---

#### 5.5.12.8. GetLineSource method

This method gets current data source of the specified output line.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public CamApiStatus GetLineSource(  
    CameraLineSelector selector,  
    out
```

---

```
CameraLineSource source
)
```

```
[VB.NET]
```

```
Public Function GetLineSource (
    selector As CameraLineSelector,
    <OutAttribute> ByRef source As
```

---

CameraLineSource  
) As [CamApiStatus](#)

[C++]  
public:  
    [CamApiStatus](#) GetLineSource(  
        [CameraLineSelector](#) selector,  
        [OutAttribute]

---

```
CameraLineSource% source
)
```

#### **[Parameters]**

Parameters	Description
<i>selector</i>	T <a href="#">arget output line</a>
<i>source</i>	A variable that receives <a href="#">data source</a> of the specified output line.

#### **[Return value]**

Returns result status.

#### **[Remarks]**

This method will return error status if “LineSelector” or “LineSource” register (or node) is not implemented in the camera.

---

#### 5.5.12.9. SetLineSource method

This method sets data source of the specified output line.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public CamApiStatus SetLineSource(  
    CameraLineSelector selector,
```

---

```
CameraLineSource source
)
```

```
[VB.NET]
```

```
Public Function SetLineSource (
    selector As CameraLineSelector,
    source As
```

---

CameraLineSource  
) As [CamApiStatus](#)

[C++]  
public:  
[CamApiStatus](#) SetLineSource(  
    [CameraLineSelector](#) selector,

---

CameraLineSource *source*  
)

**[Parameters]**

Parameters	Description
<i>selector</i>	<a href="#">Target output line</a>
<i>source</i>	New <a href="#">data source</a> of the specified output line.

**[Return value]**

Returns result status.

**[Remarks]**

This method will return error status if “LineSelector” or “LineSource” register (or node) is not implemented in the camera.



---

### 5.5.13.TimerControl methods

These methods control timer.

Timer in the camera is used to output TimerActive signal.



Methods in this section control Timer0Active, because Current BU and BG series has only one timer. Refer to instruction manual of the camera about timer.

#### 5.5.13.1. GetTimerDurationMinMax method

This method gets the maximum and minimum available period that Timer0 signal is active.

namespace: Teli.TeliCamAPI.NET

---

##### [Syntax]

[C#]

```
public CamApiStatus GetTimerDurationMinMax(  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetTimerDurationMinMax (  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTimerDurationMinMax(  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

---

##### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives the minimum available timer duration value in micro seconds.
<i>max</i>	A variable that receives the minimum available timer duration value in micro seconds.

---

##### [Return value]

Returns result status.

---

##### [Remarks]

This method will return error status if “TimerDuration” register or node is not implemented in the camera.

---

### 5.5.13.2. GetTimerDuration method

This method gets current period of Timer0 active.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetTimerDuration(  
    out double microseconds  
)
```

[VB.NET]

```
Public Function GetTimerDuration (  
    <OutAttribute> ByRef microseconds As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTimerDuration(  
    [OutAttribute] Double% microseconds  
)
```

#### [Parameters]

Parameters	Description
<i>microseconds</i>	A variable that receives current "TimerDuration" value in micro seconds.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TimerDuration" register or node is not implemented in the camera.

---

### 5.5.13.3. SetTimerDuration method

This method sets period of Timer0 active.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetTimerDuration(  
    double microseconds  
)
```

[VB.NET]

```
Public Function SetTimerDuration (  
    microseconds As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetTimerDuration(  
    Double microseconds  
)
```

#### [Parameters]

Parameters	Description
<i>microseconds</i>	New "TimerDuration" value in micro seconds.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TimerDuration" register or node is not implemented in the camera.

---

#### 5.5.13.4. GetTimerDelayMinMax method

the maximum and minimum available delay time for Timer0Active singal.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetTimerDelayMinMax(  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetTimerDelayMinMax (  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTimerDelayMinMax(  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

##### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives the minimum available timer delay value in microseconds.
<i>max</i>	A variable that receives the minimum available timer delay value in microseconds.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "TimerDelay" register or node is not implemented in the camera.

---

### 5.5.13.5. GetTimerDelay method

This method gets current delay time for Timer0Active signal

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetTimerDelay(  
    out double microseconds  
)
```

[VB.NET]

```
Public Function GetTimerDelay (  
    <OutAttribute> ByRef microseconds As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTimerDelay(  
    [OutAttribute] Double% microseconds  
)
```

#### [Parameters]

Parameters	Description
<i>microseconds</i>	A variable that receives current "TimerDelay" value in microseconds.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TimerDelay" register or node is not implemented in the camera.

---

### 5.5.13.6. SetTimerDelay method

This method sets delay time for Timer0Active signal

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetTimerDelay(  
    double microseconds  
)
```

[VB.NET]

```
Public Function SetTimerDelay (  
    microseconds As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetTimerDelay(  
    Double microseconds  
)
```

#### [Parameters]

Parameters	Description
<i>microseconds</i>	New "TimerDelay" value in microseconds.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TimerDelay" register or node is not implemented in the camera.

---

### 5.5.13.7. GetTimerTriggerSource method

This method gets current [trigger source signal](#) of Timer0.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetTimerTriggerSource(  
    out CameraTimerTriggerSource source  
)
```

[VB.NET]

```
Public Function GetTimerTriggerSource (  
    <OutAttribute> ByRef source As CameraTimerTriggerSource  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetTimerTriggerSource(  
    [OutAttribute] CameraTimerTriggerSource% source  
)
```

#### [Parameters]

Parameters	Description
<i>source</i>	A variable that receives current <a href="#">trigger source signal</a> of “Timer0Active” signal.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “TimerTriggerSource” register or node is not implemented in the camera.

---

### 5.5.13.8. SetTimerTriggerSource method

This method sets [trigger source signal](#) of Timer0.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetTimerTriggerSource(  
    CameraTimerTriggerSource source  
)
```

[VB.NET]

```
Public Function SetTimerTriggerSource (  
    source As CameraTimerTriggerSource  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetTimerTriggerSource(  
    CameraTimerTriggerSource source  
)
```

#### [Parameters]

Parameters	Description
<i>source</i>	A variable that receives current <a href="#">trigger source signal</a> of "Timer0Active" signal.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "TimerTriggerSource" register or node is not implemented in the camera.

The available [source signal](#) depends on model of the camera. Refer to instruction manual of the camera about available [timer trigger source signal](#).



---

### 5.5.14. Gain methods

These methods control Gain feature of the camera.

Refer to instruction manual of the camera about Gain.

#### 5.5.14.1. GetGainMinMax method

This method gets the maximum and minimum available Gain value.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CamApiStatus GetGainMinMax(  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetGainMinMax (  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetGainMinMax(  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

---

**[Parameters]**

Parameters	Description
<i>min</i>	A variable that receives the minimum available Gain value in dB.
<i>max</i>	A variable that receives the minimum available Gain value in dB.

---

**[Return value]**

Returns result status.

---

**[Remarks]**

This method will return error status if "Gain" register or node is not implemented in the camera.

---

### 5.5.14.2. GetGain method

This method gets current Gain value of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetGain(  
    out double gain  
)
```

[VB.NET]

```
Public Function GetGain (  
    <OutAttribute> ByRef gain As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetGain(  
    [OutAttribute] Double% gain  
)
```

#### [Parameters]

Parameters	Description
<i>gain</i>	A variable that receives current "Gain" value in dB.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "Gain" register or node is not implemented in the camera.

---

### 5.5.14.3. SetGain method

This method sets Gain of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetGain(  
    double gain  
)
```

[VB.NET]

```
Public Function SetGain (  
    gain As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetGain(  
    Double gain  
)
```

#### [Parameters]

Parameters	Description
<i>gain</i>	New "Gain" value in dB.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "Gain" register or node is not implemented in the camera.

---

#### 5.5.14.4. GetGainAuto method

This method gets current AGC mode.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetGainAuto(  
    out CameraGainAuto value  
)
```

[VB.NET]

```
Public Function GetGainAuto (  
    <OutAttribute> ByRef value As CameraGainAuto  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetGainAuto(  
    [OutAttribute] CameraGainAuto% value  
)
```

##### [Parameters]

Parameters	Description
<i>gain</i>	A variable that receives current AGC mode.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "GainAuto" register or node is not implemented in the camera.

---

### 5.5.14.5. SetGainAuto メソッド

This method sets [AGC mode](#).

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetGainAuto(  
    CameraGainAuto value  
)
```

[VB.NET]

```
Public Function SetGainAuto (  
    value As CameraGainAuto  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus SetGainAuto(  
        CameraGainAuto value  
)
```

#### [Parameters]

Parameters	Description
<i>gain</i>	New <a href="#">AGC mode</a> .

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “GainAuto” register or node is not implemented in the camera.

---

### 5.5.15.BlackLevel methods

These methods control Black-Level of the camera.

Refer to instruction manual of the camera about Black-Level control.

#### 5.5.15.1. GetBlackLevelMinMax method

This method gets the maximum and minimum available Black-Level value.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CamApiStatus GetBlackLevelMinMax(  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetBlackLevelMinMax (  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetBlackLevelMinMax(  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

---

**[Parameters]**

Parameters	Description
<i>min</i>	A variable that receives the minimum available Black-Level value in %.
<i>max</i>	A variable that receives the minimum available Black-Level value in %.

---

**[Return value]**

Returns result status.

---

**[Remarks]**

This method will return error status if "BlackLevel" register or node is not implemented in the camera.

---

### 5.5.15.2. GetBlackLevel method

This method gets current Black-Level value of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetBlackLevel(  
    out double blackLevel  
)
```

[VB.NET]

```
Public Function GetBlackLevel (  
    <OutAttribute> ByRef blackLevel As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetBlackLevel(  
    [OutAttribute] Double% blackLevel  
)
```

#### [Parameters]

Parameters	Description
<i>blackLevel</i>	A variable that receives current Black-Level value in %.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "BlackLevel" register or node is not implemented in the camera.

---

### 5.5.15.3. SetBlackLevel method

This method sets Black-Level value of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetBlackLevel(  
    double blackLevel  
)
```

[VB.NET]

```
Public Function SetBlackLevel (  
    blackLevel As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetBlackLevel(  
    Double blackLevel  
)
```

#### [Parameters]

Parameters	Description
<i>blackLevel</i>	New Black-Level value in %.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "BlackLevel" register or node is not implemented in the camera.



---

### 5.5.16. Gamma methods

These methods control Gamma correction feature of the camera.

Refer to instruction manual of the camera about Gamma correction

#### 5.5.16.1. GetGammaMinMax method

This method gets the maximum and minimum available gamma correction value.

namespace: Teli.TeliCamAPI.NET

---

##### [Syntax]

[C#]

```
public CamApiStatus GetGammaMinMax(  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetGammaMinMax (  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetGammaMinMax(  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

---

##### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives the minimum available Gamma correction value.
<i>max</i>	A variable that receives the minimum available Gamma correction value.

---

##### [Return value]

Returns result status.

---

##### [Remarks]

This method will return error status if "Gamma" register or node is not implemented in the camera.

---

### 5.5.16.2. GetGamma method

This method gets Camma correction value of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetGamma(  
    out double gamma  
)
```

[VB.NET]

```
Public Function GetGamma (  
    <OutAttribute> ByRef gamma As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetGamma(  
    [OutAttribute] Double% gamma  
)
```

#### [Parameters]

Parameters	Description
<i>gain</i>	A variable that receives current Gamma correction value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "Gamma" register or node is not implemented in the camera.

---

### 5.5.16.3. SetGamma method

This method sets Gamma correction value of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetGamma(  
    double gamma  
)
```

[VB.NET]

```
Public Function SetGamma (  
    gamma As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetGamma(  
    Double gamma  
)
```

#### [Parameters]

Parameters	Description
<i>gain</i>	New Gamma correction value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "Gamma" register or node is not implemented in the camera.

---

### 5.5.17.WhiteBalance methods

These methods control White balance ratio of the camera.

These methods are available only in color model cameras.

Camera will control White balance, adjusting relative gain value of R/G and B/G, manually or automatically.

Refer to instruction manual of the camera about White balance.

#### 5.5.17.1. GetBalanceRatioMinMax method

This method gets the maximum and minimum available White balance gain value.

namespace: Teli.TeliCamAPI.NET

---

##### [Syntax]

[C#]

```
public CamApiStatus GetBalanceRatioMinMax(  
    CameraBalanceRatioSelector selector,  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetBalanceRatioMinMax (  
    selector As CameraBalanceRatioSelector,  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetBalanceRatioMinMax(  
    CameraBalanceRatioSelectorselector,  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

---

##### [Parameters]

Parameters	Description
<i>selector</i>	<a href="#">Target color component</a> (R or B).
<i>min</i>	A variable that receives the minimum available Gain value.
<i>max</i>	A variable that receives the minimum available Gaain value.

---

##### [Return value]

Returns result status.

---

##### [Remarks]

This method will return error status if “BalanceRatioSelector” or “BalanceRatio” register (or node) is not implemented in the camera.

---

### 5.5.17.2. GetBalanceRatio method

This method gets current White balance relative gain value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetBalanceRatio(  
    CameraBalanceRatioSelector selector,  
    out double value  
)
```

[VB.NET]

```
Public Function GetBalanceRatio (  
    selector As CameraBalanceRatioSelector,  
    <OutAttribute> ByRef value As Double  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetBalanceRatio(  
        CameraBalanceRatioSelector selector,  
        [OutAttribute] Double% value  
)
```

#### [Parameters]

Parameters	Description
<i>selector</i>	<a href="#">Target color component</a> (R or B).
<i>value</i>	A variable that receives current relative gain value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “BalanceRatioSelector” or “BalanceRatio” register (or node) is not implemented in the camera.

---

### 5.5.17.3. SetBalanceRatio method

This method sets White balance gain value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetBalanceRatio(  
    CameraBalanceRatioSelector selector,  
    ref double value  
)
```

[VB.NET]

```
Public Function SetBalanceRatio (  
    selector As CameraBalanceRatioSelector,  
    ByRef value As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetBalanceRatio(  
    CameraBalanceRatioSelector selector,  
    Double% value  
)
```

#### [Parameters]

Parameters	Description
<i>selector</i>	<a href="#">Target color component</a> (R or B).
<i>value</i>	New relative gain value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “BalanceRatioSelector” or “BalanceRatio” register (or node) is not implemented in the camera.

---

#### 5.5.17.4. GetBalanceWhiteAuto method

This method gets current automatic white balance mode.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetBalanceWhiteAuto(  
    out CameraBalanceWhiteAuto value  
)
```

[VB.NET]

```
Public Function GetBalanceWhiteAuto (  
    <OutAttribute> ByRef value As CameraBalanceWhiteAuto  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetBalanceWhiteAuto(  
    [OutAttribute] CameraBalanceWhiteAuto% value  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current <a href="#">automatic white balance mode</a> .

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “BalanceWhiteAuto” register or node is not implemented in the camera.

---

### 5.5.17.5. SetBalanceWhiteAuto method

This method sets automaticwhite balance mode.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetBalanceWhiteAuto(  
    CameraBalanceWhiteAuto value  
)
```

[VB.NET]

```
Public Function SetBalanceWhiteAuto (  
    value As CameraBalanceWhiteAuto  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetBalanceWhiteAuto(  
    CameraBalanceWhiteAuto value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New <a href="#">automatic white balance mode</a> .

#### [Return value]

Returns result status.

#### [Remarks]

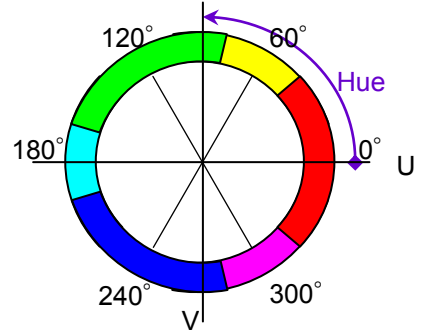
This method will return error status if “BalanceWhiteAuto” register or node is not implemented in the camera.



---

### 5.5.18.Hue methods

These methods control Hue adjustment feature of the camera.  
This function is available only in color model cameras.  
Refer to instruction manual of the camera about Hue.



#### 5.5.18.1. GetHueMinMax method

This method gets the maximum and minimum available hue value.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

```
[C#]
public CamApiStatus GetHueMinMax(
    out double min,
    out double max
)
```

##### [VB.NET]

```
Public Function GetHueMinMax (
    <OutAttribute> ByRef min As Double,
    <OutAttribute> ByRef max As Double
) As CamApiStatus
```

##### [C++]

```
public:
CamApiStatus GetHueMinMax(
    [OutAttribute] Double% min,
    [OutAttribute] Double% max
)
```

##### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives the minimum available Hue value in degrees.
<i>max</i>	A variable that receives the minimum available Hue value in degrees.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “Hue” register or node is not implemented in the camera.

---

### 5.5.18.2. GetHue method

This method gets current Hue correction value of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetHue(  
    out double hue  
)
```

[VB.NET]

```
Public Function GetHue (  
    <OutAttribute> ByRef hue As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetHue(  
    [OutAttribute] Double% hue  
)
```

#### [Parameters]

Parameters	Description
<i>hue</i>	A variable that receives current Hue correction value in degrees.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "Hue" register or node is not implemented in the camera.

---

### 5.5.18.3. SetHue method

This method sets Hue correction value of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetHue(  
    double hue  
)
```

[VB.NET]

```
Public Function SetHue (  
    hue As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetHue(  
    Double hue  
)
```

#### [Parameters]

Parameters	Description
<i>hue</i>	New Hue correction value in degrees.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "Hue" register or node is not implemented in the camera.

---

### 5.5.19.Saturation methods

These methods control Saturation feature of the camera.  
These methods are available only in color model cameras.  
Refer to instruction manual of the camera about Saturation.

#### 5.5.19.1. GetSaturationMinMax method

This method gets the maximum and minimum available saturation vaue.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetSaturationMinMax(  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetSaturationMinMax (  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSaturationMinMax(  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

##### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives the minimum available Saturation value.
<i>max</i>	A variable that receives the minimum available Saturation value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "Saturation" register or node is not implemented in the camera.

---

### 5.5.19.2. GetSaturation method

This method gets current saturation value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetSaturation(  
    out double saturation  
)
```

[VB.NET]

```
Public Function GetSaturation (  
    <OutAttribute> ByRef saturation As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSaturation(  
    [OutAttribute] Double% saturation  
)
```

#### [Parameters]

Parameters	Description
<i>saturation</i>	A variable that receives current Saturation value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "Saturation" register or node is not implemented in the camera.

---

### 5.5.19.3. SetSaturation method

This method sets saturation value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetSaturation(  
    double saturation  
)
```

[VB.NET]

```
Public Function SetSaturation (  
    saturation As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetSaturation(  
    Double saturation  
)
```

#### [Parameters]

Parameters	Description
<i>saturation</i>	New Saturation value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "Saturation" register or node is not implemented in the camera.

---

### 5.5.20. Sharpness methods

These methods control sharpness feature of the camera.

Refer to instruction manual of the camera about Sharpness.

#### 5.5.20.1. GetSharpnessMinMax method

This method gets the maximum and minimum available sharpness feature value.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

```
[C#]  
public CamApiStatus GetSharpnessMinMax(  
    out int min,  
    out int max  
)
```

##### [VB.NET]

```
Public Function GetSharpnessMinMax (  
    <OutAttribute> ByRef min As Integer,  
    <OutAttribute> ByRef max As Integer  
) As CamApiStatus
```

##### [C++]

```
public:  
CamApiStatus GetSharpnessMinMax(  
    [OutAttribute] Int32% min,  
    [OutAttribute] Int32% max  
)
```

##### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives the minimum available Sharpness value.
<i>max</i>	A variable that receives the minimum available Sharpness value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “Sharpness” register or node is not implemented in the camera.

---

### 5.5.20.2. GetSharpness method

This method gets current sharpness correction value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetSharpness(  
    out int sharpness  
)
```

[VB.NET]

```
Public Function GetSharpness (  
    <OutAttribute> ByRef sharpness As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSharpness(  
    [OutAttribute] Int32% sharpness  
)
```

#### [Parameters]

Parameters	Description
<i>sharpness</i>	A variable that receives current Sharpness correction value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “Sharpness” register or node is not implemented in the camera.



---

### 5.5.20.3. SetSharpness method

This method sets sharpness correction value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetSharpness(  
    int sharpness  
)
```

[VB.NET]

```
Public Function SetSharpness (  
    sharpness As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetSharpness(  
    Int32 sharpness  
)
```

#### [Parameters]

Parameters	Description
<i>sharpness</i>	New Sharpness correction value.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “Sharpness” register or node is not implemented in the camera.

---

### 5.5.21. ColorCorrectionMatrix methods

These methods control Color Correction Matrix for correcting pixel color value.

The relationship between original pixel data (R, G, and B) and corrected pixel data (R', G', and B') are represented in the following formula.

$$\begin{pmatrix} R' \\ G' \\ B' \end{pmatrix} = \begin{pmatrix} 1 & -mask\_rg & -mask\_rb \\ -mask\_gr & 1 & -mask\_gb \\ -mask\_br & -mask\_bg & 1 \end{pmatrix} \begin{pmatrix} R & (G-R) & (B-R) \\ (R-G) & G & (B-G) \\ (R-B) & (G-B) & B \end{pmatrix}$$
$$R' = (1 - mask\_rg - mask\_rb) \times R + mask\_rg \times G + mask\_rb \times B$$
$$G' = mask\_gr \times R + (1 - mask\_gr - mask\_gb) \times G + mask\_gb \times B$$
$$B' = mask\_br \times R + mask\_bg \times G + (1 - mask\_br - mask\_bg) \times B$$

GenApi module uses selectors, "SelectorI" and "SelectorJ", for selecting matrix element.

The correspondence of "SelectorI" and "SelectorJ" to color correction matrix element is as follows.

	SelectorJ = R	SelectorJ = G	SelectorJ = B
SelectorI = R		mask_rg	mask_rb
SelectorI = G	mask_gr		mask_gb
SelectorI = B	mask_br	mask_bg	

The following methods use "CameraColorCorrectionMatrixSelector" type element selector, that specifies both "SelectorI" and "SelectorJ" in an enumeration member value.

Refer to instruction manual of the camera about ColorCorrectionMatrix.

#### 5.5.21.1. GetColorCorrectionMatrixMinMax method

This method gets the maximum and minimum coefficient of color correction matrix.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetColorCorrectionMatrixMinMax(  
    CameraColorCorrectionMatrixSelector selector,  
    out double min,  
    out double max  
)
```

[VB.NET]

```
Public Function GetColorCorrectionMatrixMinMax (  
    selector As CameraColorCorrectionMatrixSelector,  
    <OutAttribute> ByRef min As Double,  
    <OutAttribute> ByRef max As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetColorCorrectionMatrixMinMax(  
    CameraColorCorrectionMatrixSelector selector,  
    [OutAttribute] Double% min,  
    [OutAttribute] Double% max  
)
```

---

**[Parameters]**

---

Parameters	Description
<i>selector</i>	<a href="#">Target element of Color CorrectionMatrix.</a>
<i>min</i>	A variable that receives the minimum available coefficient value.
<i>max</i>	A variable that receives the minimum available coefficient value.

**[Return value]**

---

Returns result status.

**[Remarks]**

---

This method will return error status if “ColorCorrectionMatrix”, “ColorCorrectionMatrixSelect orI”, and “ColorCorrectionMatrixJ” registers (or nodes) are not implemented in the camera.

---

### 5.5.21.2. GetColorCorrectionMatrix method

This method gets current coefficient of specified Color Correction Matrix element.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetColorCorrectionMatrix(  
    CameraColorCorrectionMatrixSelector selector,  
    out double matrix  
)
```

[VB.NET]

```
Public Function GetColorCorrectionMatrix (  
    selector As CameraColorCorrectionMatrixSelector,  
    <OutAttribute> ByRef matrix As Double  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetColorCorrectionMatrix(  
        CameraColorCorrectionMatrixSelector selector,  
        [OutAttribute] Double% matrix  
)
```

#### [Parameters]

Parameters	Description
<i>selector</i>	<a href="#">Target element of Color CorrectionMatrix.</a>
<i>sharpness</i>	A variable that receives current coefficient of the element.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “ColorCorrectionMatrix”, “ColorCorrectionMatrixSelectorI”, and “ColorCorrectionMatrixJ” registers (or nodes) are not implemented in the camera.

---

### 5.5.21.3. SetColorCorrectionMatrix method

This method sets coefficient of specified Color Correction Matrix element.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetColorCorrectionMatrix(  
    CameraColorCorrectionMatrixSelector selector,  
    double matrix  
)
```

[VB.NET]

```
Public Function SetColorCorrectionMatrix (  
    selector As CameraColorCorrectionMatrixSelector,  
    matrix As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetColorCorrectionMatrix(  
    CameraColorCorrectionMatrixSelector selector,  
    Double matrix  
)
```

#### [Parameters]

Parameters	Description
<i>selector</i>	<a href="#">Target element of Color CorrectionMatrix.</a>
<i>sharpness</i>	New coefficient of the element.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “ColorCorrectionMatrix”, “ColorCorrectionMatrixSelect orI”, and “ColorCorrectionMatrixJ” registers (or nodes) are not implemented in the camera.

---

## 5.5.22.LUT Control methods

These methods control LUT (Look Up Table) feature for correcting pixel value.

Refer to instruction manual of the camera about LUT.

### 5.5.22.1. GetLutEnable method

This method gets whether LUT feature of the camera is enabled.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetLutEnable(  
    out bool value  
)
```

[VB.NET]

```
Public Function GetLutEnable (  
    <OutAttribute> ByRef value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetLutEnable(  
    [OutAttribute] Boolean % value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current state of LUT. If true, LUT is enabled.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “LUTEnable” register or node is not implemented in the camera.

---

### 5.5.22.2. SetLutEnable method

This method enables or disables LUT feature of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetLutEnable(  
    bool value  
)
```

[VB.NET]

```
Public Function SetLutEnable (  
    value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetLutEnable(  
    bool value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New state of LUT. If true, LUT will be enabled.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “LUTEnable” register or node is not implemented in the camera.

---

### 5.5.22.3. GetLutValue method

This method gets current output value of specified LUT element.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetLutValue(  
    int index,  
    out int value  
)
```

[VB.NET]

```
Public Function GetLutValue (  
    index As Integer,  
    <OutAttribute> ByRef value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetLutValue(  
    Int32 index,  
    [OutAttribute] Int32% value  
)
```

#### [Parameters]

Parameters	Description
<i>index</i>	Index of target LUT element (input value of IUT). The available index value range is from 0 up to 1023, or from 0 up to 4095.
<i>value</i>	A variable that receives current value of the LUT element (output value of UT). The available index value range is from 0 up to 1023, or from 0 up to 4095.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “LUTIndex” register and “LUTValue” (or “LUTEntry”) register (or nodes) are not implemented in the camera.

Range of input and output values of LUT, depends on the camera.



---

#### 5.5.22.4. SetLutValue method

This method sets output value of specified LUT element.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetLutValue(  
    int index,  
    int value  
)
```

[VB.NET]

```
Public Function SetLutValue (  
    index As Integer,  
    value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetLutValue(  
    Int32 index,  
    Int32 value  
)
```

#### [Parameters]

Parameters	Description
<i>index</i>	Index of target LUT element (input value of IUT). The available index value range is from 0 up to 1023, or from 0 up to 4095.
<i>value</i>	New value of the LUT element (output value of UT). The available index value range is from 0 up to 1023, or from 0 up to 4095.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “LUTIndex” register and “LUTValue” (or “LUTEntry”) register (or nodes) are not implemented in the camera.

Range of input and output values of LUT, depends on the camera.

---

### 5.5.23. UserSetControl methods

These methods control UserSet feature of the camera.

User application can save current parameters of the camera to non-volatile UserSet memory, and load saved parameters to registers of the camera.

Refer to instruction manual of the camera about UserSet feature.

#### 5.5.23.1. ExecuteUserSetLoad method

This method sends command for loading parameters stored in non-volatile UserSet memory to registers of the camera.

namespace: Teli.TeliCamAPI.NET

---

##### [Syntax]

[C#]

```
public CamApiStatus ExecuteUserSetLoad(  
    CameraUserSetSelector selector  
)
```

[VB.NET]

```
Public Function ExecuteUserSetLoad (  
    selector As CameraUserSetSelector  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus ExecuteUserSetLoad(  
        CameraUserSetSelector selector  
)
```

---

##### [Parameters]

Parameters	Description
<i>selector</i>	<a href="#">Channel of UserSet memory</a> to load parameters.

---

##### [Return value]

Returns result status.

---

##### [Remarks]

This method will return error status if “UserSetSelector” and “UserSetLoad” registers (or nodes) are not implemented in the camera.

Refer to instruction manual of the camera, to check target registers for loading from user memory.

---

### 5.5.23.2. ExecuteUserSetSave method

This method sends command for saving current parameters in non- volatile UserSet memory to the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus ExecuteUserSetSave(  
    CameraUserSetSelector selector  
)
```

[VB.NET]

```
Public Function ExecuteUserSetSave (  
    selector As CameraUserSetSelector  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus ExecuteUserSetSave(  
    CameraUserSetSelector selector  
)
```

#### [Parameters]

Parameters	Description
<i>selector</i>	<a href="#">Channel of UserSet memory</a> to save parameters. CameraUserSetSelector.Default is not available for saving target.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “UserSetSelector” and “UserSetSave” registers (or nodes) are not implemented in the camera.

Refer to instruction manual of the camera, to check target registers for saving parameters to user memory.

---

### 5.5.23.3. ExecuteUserSetQuickSave method

This method sends command for saving current parameters in volatile UserSet memory to the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus ExecuteUserSetQuickSave(  
    CameraUserSetSelector selector  
)
```

[VB.NET]

```
Public Function ExecuteUserSetQuickSave (  
    selector As CameraUserSetSelector  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus ExecuteUserSetQuickSave (  
    CameraUserSetSelector selector  
)
```

#### [Parameters]

---

Parameters	Description
<i>selector</i>	<a href="#">Channel of UserSet memory</a> to save parameters. CameraUserSetSelector.Default is not available for saving target.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

This method will return error status if “UserSetSelector” and “UserSetQuickSave” registers (or nodes) are not implemented in the camera.

UserSetQuickSave can reduce the overhed time of UserSetSave greatly because it stored to internal RAM. You can also save UserSets to non-volatile memory(Serial Flash) if necessary by UserSetSave. (backward compatible)

Refer to instruction manual of the camera, to check target registers for saving parameters to user memory. The difference between UserSetSave and UserSetQuickSave is described in instruction manual of the camera.

---

#### 5.5.23.4. GetUserSetDefault method

This method gets a user setting channel to be loaded when the camera is powered on.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetUserSetDefault (  
    out CameraUserSetSelector selector  
)
```

[VB.NET]

```
Public Function GetUserSetDefault (  
    <OutAttribute> ByRef selector As CameraUserSetSelector  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetUserSetDefault (  
    [OutAttribute] CameraUserSetSelector% selector  
)
```

##### [Parameters]

Parameters	Description
<i>selector</i>	A variable that receives <a href="#">Channel of UserSet memory</a> .

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "UserSetDefault" register (or node) is not implemented in the camera.

---

### 5.5.23.5. SetUserSetDefault method

This method sets a user setting channel to be loaded when the camera is powered on.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetUserSetDefault (  
    CameraUserSetSelector selector  
)
```

[VB.NET]

```
Public Function SetUserSetDefault (  
    selector As CameraUserSetSelector  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetUserSetDefault (  
    CameraUserSetSelector selector  
)
```

#### [Parameters]

Parameters	Description
<i>selector</i>	New <a href="#">Channel of UserSet memory</a> .

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "UserSetDefault" register (or node) is not implemented in the camera.

When you write a value to UserSetDefault register, some cameras are saved in non-volatile memory, but some cameras are not saved. (Depending on the camera's firmware version.) For cameras that are not saved in non-volatile memory, use [ExecuteUserSetSaveAndSetDefault](#) method.

Refer to instruction manual of the camera about UserSet

---

### 5.5.23.6. ExecuteUserSetSaveAndSetDefault method

This method sends command for saving current parameters in non- volatile UserSet memory and sets the specified channel as a channel loaded on initialization of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus ExecuteUserSetSaveAndSetDefault(  
    CameraUserSetSelector selector  
)
```

[VB.NET]

```
Public Function ExecuteUserSetSaveAndSetDefault (  
    selector As CameraUserSetSelector  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus ExecuteUserSetSaveAndSetDefault(  
    CameraUserSetSelector selector  
)
```

#### [Parameters]

Parameters	Description
<i>selector</i>	<a href="#">Channel of UserSet memory</a> to save parameters.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “UserSetSelector” and “UserSetLoad” registers (or nodes) are not implemented in the camera.

When other than “CameraUserSetSelector.Default” is specified to [selector](#), current parameters will be saved to the specified channel. The camera will start up with parameters saved in the specified channel from the next power-on.

When “CameraUserSetSelector.Default” is specified to [selector](#), current parameters will be discarded and initial factory parameters will be loaded immediately. The camera will start up with initial factory parameters from the next power-on.

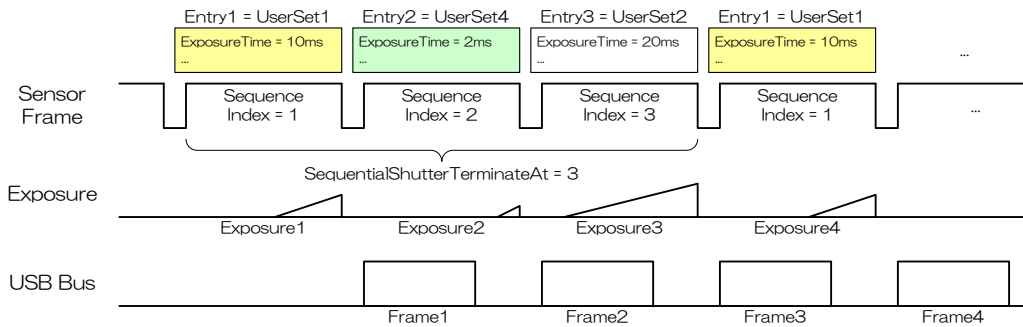
Refer to instruction manual of the camera, to check target registers for saving parameters.

### 5.5.24.SequentialShutterControl methods

These methods control Sequential Shutter feature of the camera.

Sequential Shutter feature switches acquisition parameters saved in UserSet memory sequentially.

Refer to instruction manual of the camera about Sequential shutter.



#### 5.5.24.1. GetSequentialShutterEnable method

This method gets status (enabled / disabled) of Sequential Shutter mode.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetSequentialShutterEnable(  
    out bool value  
)
```

[VB.NET]

```
Public Function GetSequentialShutterEnable (  
    <OutAttribute> ByRef value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSequentialShutterEnable(  
    [OutAttribute] Boolean% value  
)
```

##### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current state of Sequential Shutter mode. If true, Sequential Shutter is enabled.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "SequentialShutterEnable" register or node is not implemented in the camera.



---

### 5.5.24.2. SetSequentialShutterEnable method

This method enables / disables Sequential Shutter mode of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetSequentialShutterEnable(  
    bool value  
)
```

[VB.NET]

```
Public Function SetSequentialShutterEnable (  
    value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetSequentialShutterEnable(  
    Boolean value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New Sequential Shutter mode status. If true, Sequential Shutter will be enabled.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "SequentialShutterEnable" register or node is not implemented in the camera.

---

### 5.5.24.3. GetSequentialShutterTerminateAtMinMax method

This method gets the maximum and minimum available value for “SequentialShutter” sequence length.

Refer to figure in 5.5.24 SequentialShutterControl methods about SequentialShutterTerminateAt.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetSequentialShutterTerminateAtMinMax(  
    out int min,  
    out int max  
)
```

[VB.NET]

```
Public Function GetSequentialShutterTerminateAtMinMax (  
    <OutAttribute> ByRef min As Integer,  
    <OutAttribute> ByRef max As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSequentialShutterTerminateAtMinMax(  
    [OutAttribute] Int32% min,  
    [OutAttribute] Int32% max  
)
```

#### [Parameters]

---

Parameters	Description
<i>min</i>	A variable that receives the minimum available value.
<i>max</i>	A variable that receives the minimum available value.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

This method will return error status if “SequentialShutterTerminateAt” register or node is not implemented in the camera.

---

#### 5.5.24.4. GetSequentialShutterTerminateAt method

This method gets current sequence length of “SequentialShutter” mode.  
Refer to figure in 5.5.24 SequentialShutterControl methods about  
SequentialShutterTerminateAt.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public CamApiStatus GetSequentialShutterTerminateAt(  
    out int value  
)
```

[VB.NET]

```
Public Function GetSequentialShutterTerminateAt (  
    <OutAttribute> ByRef value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSequentialShutterTerminateAt(  
    [OutAttribute] Int32% value  
)
```

##### **[Parameters]**

---

Parameters	Description
<i>value</i>	A variable that receives current sequence length of “SequentialShutter” mode.

##### **[Return value]**

---

Returns result status.

##### **[Remarks]**

---

This method will return error status if “SequentialShutterTerminateAt” register or node is not implemented in the camera.

---

#### 5.5.24.5. SetSequentialShutterTerminateAt method

This method sets sequence length of “SequentialShutter” mode.  
Refer to figure in 5.5.24 SequentialShutterControl methods about SequentialShutterTerminateAt.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public CamApiStatus SetSequentialShutterTerminateAt(  
    int value  
)
```

[VB.NET]

```
Public Function SetSequentialShutterTerminateAt (  
    value As Integer  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus SetSequentialShutterTerminateAt(  
        Int32 value  
)
```

##### **[Parameters]**

---

Parameters	Description
<i>value</i>	New sequence length of “SequentialShutter” mode.

##### **[Return value]**

---

Returns result status.

##### **[Remarks]**

---

This method will return error status if “SequentialShutterTerminateAt” register or node is not implemented in the camera.

---

#### 5.5.24.6. GetSequentialShutterIndexMinMax method

This method gets the maximum and minimum available index of shutter definition data.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetSequentialShutterIndexMinMax(  
    out int min,  
    out int max  
)
```

[VB.NET]

```
Public Function GetSequentialShutterIndexMinMax (  
    <OutAttribute> ByRef min As Integer,  
    <OutAttribute> ByRef max As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSequentialShutterIndexMinMax(  
    [OutAttribute] Int32% min,  
    [OutAttribute] Int32% max  
)
```

##### [Parameters]

Parameters	Description
<i>min</i>	A variable that receives the minimum available value.
<i>max</i>	A variable that receives the minimum available value.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if "SequentialShutterIndex" register or node is not implemented in the camera.

---

#### 5.5.24.7. GetSequentialShutterEntryMinMax method

This method gets the maximum and minimum UserSet channel number available for specifying shutter data.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public CamApiStatus GetSequentialShutterEntryMinMax(  
    out int min,  
    out int max  
)
```

[VB.NET]

```
Public Function GetSequentialShutterEntryMinMax (  
    <OutAttribute> ByRef min As Integer,  
    <OutAttribute> ByRef max As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSequentialShutterEntryMinMax(  
    [OutAttribute] Int32% min,  
    [OutAttribute] Int32% max  
)
```

##### [Parameters]

---

Parameters	Description
<i>min</i>	A variable that receives the minimum available value.
<i>max</i>	A variable that receives the minimum available value.

##### [Return value]

---

Returns result status.

##### [Remarks]

---

This method will return error status if "SequentialShutterEntry" register or node is not implemented in the camera.

---

#### 5.5.24.8. GetSequentialShutterEntry method

This method gets current UserSet channel number registered in a sequential shutter entry.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetSequentialShutterEntry(  
    int index,  
    out int entry  
)
```

[VB.NET]

```
Public Function GetSequentialShutterEntry (  
    index As Integer,  
    <OutAttribute> ByRef entry As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetSequentialShutterEntry(  
    Int32 index,  
    [OutAttribute] Int32% entry  
)
```

##### [Parameters]

Parameters	Description
<i>index</i>	Index of target sequence for getting entry value.
<i>entry</i>	A variable that receives current UserSet channel number registered.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “SequentialShutterIndex” and “SequentialShutterEntry” registers (or nodes) are not implemented in the camera.

---

#### 5.5.24.9. SetSequentialShutterEntry method

This method registers UserSet channel number to specified sequential shutter entry.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus SetSequentialShutterEntry(  
    int index,  
    int entry  
)
```

[VB.NET]

```
Public Function SetSequentialShutterEntry (  
    index As Integer,  
    entry As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetSequentialShutterEntry(  
    Int32 index,  
    Int32 entry  
)
```

##### [Parameters]

Parameters	Description
<i>index</i>	Index of target sequence for setting entry value.
<i>entry</i>	New UserSet channel number.

##### [Return value]

Returns result status.

##### [Remarks]

This method will return error status if “SequentialShutterIndex” and “SequentialShutterEntry” registers (or nodes) are not implemented in the camera.



---

### 5.5.25. UserDefinedName (DeviceUserID) method

This method controls user defined name ("UserDefinedName" or "DeviceUserID" register) of the camera.

User application can set any string to "UserDefinedName" or "DeviceUserID" register for identifying it. The register value is stored in nonVolatile memory of the camera,

Refer to instruction manual of the camera about "UserDefinedName" or "DeviceUserID" register.

#### 5.5.25.1. GetUserDefinedName method

This method gets user defined name of the camera.

namespace: Teli.TeliCamAPI.NET

---

##### [Syntax]

[C#]

```
public CamApiStatus GetUserDefinedName(  
    out string name  
)
```

[VB.NET]

```
Public Function GetUserDefinedName (  
    <OutAttribute> ByRef name As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetUserDefinedName(  
    [OutAttribute] String^% name  
)
```

---

##### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current user defined name.

---

##### [Return value]

Returns result status.

---

##### [Remarks]

This method will return error status if either "UserDefineName" or "DeviceUserID" register (or node) is not implemented in the camera.

If user defined name registered in the camera is not "NULL terminated" , the last character may be ignored.

When user application writes new user defined name, the old user defined name may remain until "[GetNumOfCameras\(\)](#)" of CameraSystem object is called.

---

### 5.5.25.2. SetUserDefinedName method

This method sets user defined name of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetUserDefinedName(  
    string name  
)
```

[VB.NET]

```
Public Function SetUserDefinedName (  
    name As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetUserDefinedName(  
    String^ name  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New user defined name.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if either "UserDefineName" or "DeviceUserID" register (or node) is not implemented in the camera.

The length of user defined name register in GigE Vision camera is 16 bytes, that in USB3 Vision camera is 64 bytes.

This method writes user defined name as NULL terminated string, which means that the length of the name, that user can specify as user defined name, is register length minus 1.

When user application writes new user defined name, the old user defined name may remain until "[GetNumOfCameras\(\)](#)" of CameraSystem object is called.

---

## 5.5.26.Chunk methods

This method group performs control of Chunk feature.

Chunk is a function to add various information to payload data.

When the chunk output is valid, chunk data is added to the payload data after the image data.

For detailed information on chunk, please to instruction manual of the camera.

When using GenICam to acquire chunk data, it is necessary to attach the buffer storing the payload data to GenICam's chunk port.

For details, see the description of the [ChunkAttachBuffer](#) method.

### 5.5.26.1. GetChunkModeActive method

This method gets current status (enabled /disabled) of chunk data output mode of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetChunkModeActive (  
    out bool value  
)
```

[VB.NET]

```
Public Function GetChunkModeActive (  
    <OutAttribute> ByRef value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetChunkModeActive (  
    [OutAttribute] Boolean% value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	A variable that receives current state of chunk data output mode If true, chunk data output mode is enabled.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if "ChunkModeActive" register or node is not implemented in the camera.

---

### 5.5.26.2. SetChunkModeActive method

This method enables or disables chunk data output mode of the camera.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetChunkModeActive (  
    bool value  
)
```

[VB.NET]

```
Public Function SetChunkModeActive (  
    value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetChunkModeActive (  
    Boolean value  
)
```

#### [Parameters]

Parameters	Description
<i>value</i>	New state of chunk data output mode. If true, chunk data output mode will be enabled.

#### [Return value]

Returns result status.

#### [Remarks]

This method will return error status if “ChunkModeActive” register or node is not implemented in the camera.

---

## 5.5.27.Properties

### 5.5.27.1. Parent property

This property gets parent [CameraDevice](#) object.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CameraDevice Parent { get; }
```

[VB.NET]

```
Public ReadOnly Property Parent As CameraDevice  
    Get
```

[C++]

```
public:  
property CameraDevice^ Parent {  
    CameraDevice^ get ();  
}
```

---

## 5.6. GenApiWrapper class

GenApiWrapper is a class for wrapping GenICam GenApi library, which allow user application to access various information of camera register easier.

Refer to <http://www.genicam.org> about detail information of GenICam.

GenApiWrapper object will be created as a member of [CameraDevice](#) object on creating [CameraDevice](#) class instance, and is under the management of [CameraDevice](#) class.

User application should not dispose (delete) GenApiWrapper object.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

[C#]

```
public class GenApiWrapper : IDisposable
```

[VB.NET]


















```
Public Class GenApiWrapper  
    Implements IDisposable
```























[C++]

```
public ref class GenApiWrapper : IDisposable
```


The followings are members of GenApiWrapper class available for user applications.

### [Methods]

	Name	Description
INode node		
	GetNode	Gets a node object that has specified node name.
	GetNodeType	Gets node type of the specified node.
	GetName	Gets name of the node.
	GetAccessMode	Gets access mode of the node
	GetVisibility	Gets visibility of the node.
	GetCachingMode	Gets caching mode of the node.
	GetDescription	Gets description of the node.
	GetToolTip	Gets tool tip (short description) of the node.
	GetRepresentation	Gets recommended representation of the node.
	GetUnit	Gets name of the node value unit.
ICategory node		
	GetNumOfFeatures	Gets number of features that the node contains.
	GetFeatureByIndex	Gets a node object specified by index in feature list of the parent category node.
Integer node		
	GetIntMin	Gets the minimum valid value of the node.
	GetIntMax	Gets the maximum valid value of the node.
	GetIntInc	Gets the Increment value of the node.
	GetIntValue	Gets current value of the node.
	SetIntValue	Sets value to the node.

	Name	Description
<b>IFloat node</b>		
	GetFloatMin	Gets the minimum valid value of the node.
	GetFloatMax	Gets the maximum valid value of the node.
	GetFloatHasInc	Gets whether this node has Increment value.
	GetFloatInc	Gets the Increment value of the node.
	GetFloatDisplayNotation	Gets the display notation for the node value.
	GetFloatDisplayPrecision	Gets the display precision for the node value.
	GetFloatValue	Gets current value of the node.
	SetFloatValue	Sets value to the node.
<b>IBoolean node</b>		
	GetBoolValue	Gets current value of the node.
	SetBoolValue	Sets value to the node.
<b>IEnumeration node</b>		
	GetNumOfEnumEntries	Gets number of EnumEntries of the IEnumeration node.
	GetEnumIntValue	Gets current value of the IEnumeration node as an integer value.
	SetEnumIntValue	Sets value of the IEnumeration node specifying integer value.
	GetEnumStrValue	Gets current value of the IEnumeration node as a string value.
	SetEnumStrValue	Sets value of the IEnumeration node specifying string value.
	GetEnumEntryByIndex	Gets an IEnumEntry node object specified by index in EnumEntry list of the IEnumeration node.
<b>IEnumEntry node</b>		
	GetEnumEntryIntValue	Gets the IEnumEntry node value as an integer value.
	GetEnumEntryStrValue	Gets the IEnumEntry node value as a string value.
<b>ICommand node</b>		
	ExecuteCommand	Sends command of the ICommand type node.
	IsCommandDone	Gets the status whether the command has finished or not,
<b>IString node</b>		
	GetStrValue	Gets string value of the node.
	SetStrValue	Sets string value to the node.

#### [Properties]

	Name	Description
	Parent	Gets parent <a href="#">CameraDevice</a> object.

---

### 5.6.1. INode methods

INode class is a base class of all other node classes.

Methods of INode interface are available in all other node objects.

#### 5.6.1.1. GetNode method

This method gets a node object that has specified node name.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetNode(  
    string name,  
    ref GenApiNode node  
)
```

[VB.NET]

```
Public Function GetNode (  
    name As String,  
    ByRef node As GenApiNode  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetNode(  
    String^ name,  
    GenApiNode ^% node  
)
```

##### [Parameters]

Parameters	Description
<i>name</i>	Node name.
<i>node</i>	A variable that receives INode object that has specified name.

##### [Return value]

Returns result status.

##### [Remarks]

GenICam SFNC (Standard Feature Name Convention) defines standard node names for general features.

There also are vendor unique node names for vendor unique features.

Refer to instruction manual of the camera about available feature name (node name).



---

### 5.6.1.2. GetNodeType method

This function gets [node type](#) of the specified node

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetNodeType(  
    GenApiNode node,  
    out NodeType nodeType  
)
```

[VB.NET]

```
Public Function GetNodeType (  
    node As GenApiNode,  
    <OutAttribute> ByRef nodeType As NodeType  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetNodeType(  
    GenApiNode^ node,  
    [OutAttribute] NodeType% nodeType  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>nodeType</i>	A variable that receives <a href="#">node type</a> .

#### [Return value]

Returns result status.

---

### 5.6.1.3. GetName method

This method gets name of the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetName(  
    GenApiNode node,  
    out string nodeName  
)
```

[VB.NET]

```
Public Function GetName (  
    node As GenApiNode,  
    <OutAttribute> ByRef nodeName As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetName (  
    GenApiNode^ node,  
    [OutAttribute] String^% nodeName  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>nodeName</i>	A variable that receives node name.

#### [Return value]

Returns result status.

---

#### 5.6.1.4. GetAccessMode method

This method gets [access mode](#) of the node

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public CamApiStatus GetAccessMode(  
    GenApiNode node,  
    out NodeAccessMode accessMode  
)
```

[VB.NET]

```
Public Function GetAccessMode (  
    node As GenApiNode,  
    <OutAttribute> ByRef accessMode As NodeAccessMode  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetAccessMode (  
    GenApiNode^ node,  
    [OutAttribute] NodeAccessMode% accessMode  
)
```

##### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>accessMode</i>	A variable that receives <a href="#">access mode</a> of the node.

##### [Return value]

---

Returns result status.

---

### 5.6.1.5. GetVisibility method

This method gets [recommended visibility](#) of the node.

[Visibility](#) indicates desirable user level for accessing the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetVisibility (  
    GenApiNode node,  
    out NodeVisibility visibility  
)
```

[VB.NET]

```
Public Function GetVisibility (  
    node As GenApiNode,  
    <OutAttribute> ByRef visibility As NodeVisibility  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetVisibility (  
        GenApiNode^ node,  
        [OutAttribute] NodeVisibility% visibility  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>visibility</i>	A variable that receives <a href="#">recommended visibility</a> of the node.

#### [Return value]

---

Returns result status.

---

### 5.6.1.6. GetCachingMode method

This method gets [caching mode](#) of the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetCachingMode(  
    GenApiNode node,  
    out NodeCachingMode cachingMode  
)
```

[VB.NET]

```
Public Function GetCachingMode (  
    node As GenApiNode,  
    <OutAttribute> ByRef cachingMode As NodeCachingMode  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetCachingMode (  
        GenApiNode^ node,  
        [OutAttribute] NodeCachingMode% cachingMode  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>cachingMode</i>	A variable that receives <a href="#">caching mode</a> of the node.

#### [Return value]

---

Returns result status.

---

### 5.6.1.7. GetDescription method

This method gets description of the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetDescription (  
    GenApiNode node,  
    out string description  
)
```

[VB.NET]

```
Public Function GetDescription (  
    node As GenApiNode,  
    <OutAttribute> ByRef description As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetDescription(  
    GenApiNode^ node,  
    [OutAttribute] String^% description  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>description</i>	A variable that receives description of the node.

#### [Return value]

Returns result status.

---

### 5.6.1.8. GetToolTip method

This method gets tool tip (short description) of the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetToolTip(  
    GenApiNode node,  
    out string toolTip  
)
```

[VB.NET]

```
Public Function GetToolTip (  
    node As GenApiNode,  
    <OutAttribute> ByRef toolTip As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetToolTip(  
    GenApiNode^ node,  
    [OutAttribute] String^% toolTip  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>toolTip</i>	A variable that receives tool tip (short description) of the node.

#### [Return value]

---

Returns result status.

---

### 5.6.1.9. GetRepresentation method

This method gets [recommended representation](#) of the node value.

[Representation](#) is a hint about how to display value of the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetRepresentation(  
    GenApiNode node,  
    out NodeRepresentation representation  
)
```

[VB.NET]

```
Public Function GetRepresentation (  
    node As GenApiNode,  
    <OutAttribute> ByRef representation As NodeRepresentation  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetRepresentation(  
    GenApiNode^ node,  
    [OutAttribute] NodeRepresentation% representation  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>representation</i>	A variable that receives <a href="#">representation</a> of the node.

#### [Return value]

---

Returns result status.



---

### 5.6.1.10. GetUnit method

This method returns name of the node value unit.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetUnit(  
    GenApiNode node,  
    out string unit  
)
```

[VB.NET]

```
Public Function GetUnit (  
    node As GenApiNode,  
    <OutAttribute> ByRef unit As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetUnit(  
    GenApiNode^ node,  
    [OutAttribute] String^% unit  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>unit</i>	A variable that receives name of the node value unit.

#### [Return value]

---

Returns result status.

---

## 5.6.2. ICategory node

### 5.6.2.1. GetNumOfFeatures method

This method gets number of features that the specified ICategory node contains.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetNumOfFeatures(  
    GenApiNode node,  
    out int num  
)
```

[VB.NET]

```
Public Function GetNumOfFeatures (  
    node As GenApiNode,  
    <OutAttribute> ByRef num As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetNumOfFeatures (  
    GenApiNode^ node,  
    [OutAttribute] Int32% num  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>num</i>	A variable that receives number of features.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

This method is available only for ICategory node.

This method will return error status if node type of target node is not ICategory type.

---

### 5.6.2.2. GetFeaturesByIndex method

This method gets a node object specified by index in feature list of the parent category node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetFeatureByIndex(  
    GenApiNode parentCategoryNode,  
    int featureIndex,  
    ref GenApiNode featureNode  
)
```

[VB.NET]

```
Public Function GetFeatureByIndex (  
    parentCategoryNode GenApiNode,  
    featureIndex As Integer,  
    ByRef featureNode As GenApiNode  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetFeatureByIndex (  
    GenApiNode^ parentCategoryNode,  
    Int32 featureIndex,  
    GenApiNode^% featureNode  
)
```

#### [Parameters]

---

Parameters	Description
<i>parentCategoryNode</i>	Parent Category node object.
<i>featureIndex</i>	Index of specified feature in feature list of ICategory node.
<i>featureNode</i>	A variable that receives feature node object.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

This method is available only for ICategory node.

This method will return error status if node type of target node is not ICategory type.

---

### 5.6.3. Integer node

#### 5.6.3.1. GetIntMin method

This method gets the minimum valid value of the node.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public CamApiStatus GetIntMin(  
    GenApiNode node,  
    out long min  
)
```

[VB.NET]

```
Public Function GetIntMin (  
    node As GenApiNode,  
    <OutAttribute> ByRef min As Long  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetIntMin(  
    GenApiNode^ node,  
    [OutAttribute] Int64% min  
)
```

##### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>min</i>	A variable that receives the minimum valid value of the node.

##### [Return value]

---

Returns result status.

##### [Remarks]

---

This method is available only for IInteger node.

This method will return error status if node type of target node is not IInteger type.

---

### 5.6.3.2. GetIntMax method

This method gets the maximum valid value of the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetIntMax(  
    GenApiNode node,  
    out long max  
)
```

[VB.NET]

```
Public Function GetIntMax (  
    node As GenApiNode,  
    <OutAttribute> ByRef max As Long  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetIntMax (  
        GenApiNode^ node,  
        [OutAttribute] Int64% max  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>max</i>	A variable that receives the maximum valid value of the node.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for IInteger node.

This method will return error status if node type of target node is not IInteger type.

---

### 5.6.3.3. GetIntInc method

This method gets the Increment value of the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetIntInc(  
    GenApiNode node,  
    out long inc  
)
```

[VB.NET]

```
Public Function GetIntInc (  
    node As GenApiNode,  
    <OutAttribute> ByRef inc As Long  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetIntInc (  
        GenApiNode^ node,  
        [OutAttribute] Int64% inc  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>inc</i>	A variable that receives the Increment value of the node.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for IInteger node.

This method will return error status if node type of target node is not IInteger type.

---

#### 5.6.3.4. GetIntValue method

This method gets current value of the node.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetIntValue(  
    GenApiNode node,  
    out long value  
)
```

[VB.NET]

```
Public Function GetIntValue(  
    node As GenApiNode,  
    <OutAttribute> ByRef valueAs Long  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetIntValue(  
    GenApiNode^ node,  
    [OutAttribute] Int64% value  
)
```

##### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>value</i>	A variable that receives current value of the node.

##### [Return value]

Returns result status.

##### [Remarks]

This method is available only for IInteger node.

This method will return error status if node type of target node is not IInteger type.

---

### 5.6.3.5. SetIntValue method

This method sets value to the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetIntValue(  
    GenApiNode node,  
    long value  
)
```

[VB.NET]

```
Public Function SetIntValue (  
    node As GenApiNode,  
    value As Long  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetIntValue(  
    GenApiNode^ node,  
    Int64 value  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>value</i>	New value of the node.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for IInteger node.

This method will return error status if node type of target node is not IInteger type.



---

## 5.6.4. IFloat node

### 5.6.4.1. GetFloatMin method

This method gets the minimum valid value of the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetFloatMin(  
    GenApiNode node,  
    out double min  
)
```

[VB.NET]

```
Public Function GetFloatMin (  
    node As GenApiNode,  
    <OutAttribute> ByRef min As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetFloatMin(  
    GenApiNode^ node,  
    [OutAttribute] Double% min  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>min</i>	A variable that receives the minimum valid value of the node.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

This method is available only for IFloat node.

This method will return error status if node type of target node is not IFloat type.

---

#### 5.6.4.2. GetFloatMax method

This method gets the maximum valid value of the node.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetFloatMax(  
    GenApiNode node,  
    out double min  
)
```

[VB.NET]

```
Public Function GetFloatMax (  
    node As GenApiNode,  
    <OutAttribute> ByRef min As Double  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetFloatMax (  
        GenApiNode^ node,  
        [OutAttribute] Double% min  
)
```

##### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>max</i>	A variable that receives the maximum valid value of the node.

##### [Return value]

Returns result status.

##### [Remarks]

This method is available only for IFloat node.

This method will return error status if node type of target node is not IFloat type.

---

### 5.6.4.3. GetFloatHasInc method

This method gets whether this node has Increment value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetFloatHasInc(  
    GenApiNode node,  
    out bool hasInc  
)
```

[VB.NET]

```
Public Function GetFloatHasInc (  
    node As GenApiNode,  
    <OutAttribute> ByRef hasInc As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetFloatHasInc(  
        GenApiNode^ node,  
        [OutAttribute] Boolean% hasInc  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>hasInc</i>	A variable that receives whether this node has Increment value.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for IFloat node.

This method will return error status if node type of target node is not IFloat type.

---

#### 5.6.4.4. GetFloatInc method

This method gets the Increment value of the node.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetFloatInc(  
    GenApiNode node,  
    out double inc  
)
```

[VB.NET]

```
Public Function GetFloatInc (  
    node As GenApiNode,  
    <OutAttribute> ByRef inc As Double  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetFloatInc(  
        GenApiNode^ node,  
        [OutAttribute] double% inc  
    )
```

##### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>inc</i>	A variable that receives the Increment value of the node.

##### [Return value]

Returns result status.

##### [Remarks]

This method is available only for IFloat node.

This method will return error status if node type of target node is not IFloat type.

---

#### 5.6.4.5. GetFloatDisplayNotation method

This method gets the display notation for the node value.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetFloatDisplayNotation(  
    GenApiNode node,  
    out NodeFloatNotation notation  
)
```

[VB.NET]

```
Public Function GetFloatDisplayNotation (  
    node As GenApiNode,  
    <OutAttribute> ByRef notation As NodeFloatNotation  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetFloatDisplayNotation(  
        GenApiNode^ node,  
        [OutAttribute] NodeFloatNotation% notation  
)
```

##### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>notation</i>	A variable that receives the display notation for the node value.

##### [Return value]

Returns result status.

##### [Remarks]

This method is available only for IFloat node.

This method will return error status if node type of target node is not IFloat type.

---

#### 5.6.4.6. GetFloatDisplayPrecision method

This method gets the display precision for the node value.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetFloatDisplayPrecision(  
    GenApiNode node,  
    out long precision  
)
```

[VB.NET]

```
Public Function GetFloatDisplayPrecision (  
    node As GenApiNode,  
    <OutAttribute> ByRef precision As Long  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetFloatDisplayPrecision(  
        GenApiNode^ node,  
        [OutAttribute] Int64% precision  
)
```

##### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>precision</i>	A variable that receives the display precision for the node value.

##### [Return value]

Returns result status.

##### [Remarks]

This method is available only for IFloat node.

This method will return error status if node type of target node is not IFloat type.

---

#### 5.6.4.7. GetFloatValue method

This method gets current value of the node.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetFloatValue(  
    GenApiNode node,  
    out double value  
)
```

[VB.NET]

```
Public Function GetFloatValue (  
    node As GenApiNode,  
    <OutAttribute> ByRef value As Double  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus GetFloatValue(  
        GenApiNode^ node,  
        [OutAttribute] Double% value  
)
```

##### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>value</i>	A variable that receives current value of the node.

##### [Return value]

Returns result status.

##### [Remarks]

This method is available only for IFloat node.

This method will return error status if node type of target node is not IFloat type.

---

#### 5.6.4.8. SetFloatValuemethod

This method sets value to the node.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus SetFloatValue(  
    GenApiNode node,  
    double value  
)
```

[VB.NET]

```
Public Function SetFloatValue (  
    node As GenApiNode,  
    value As Double  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetFloatValue(  
    GenApiNode^ node,  
    Double value  
)
```

##### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>value</i>	New value of the node.

##### [Return value]

Returns result status.

##### [Remarks]

This method is available only for IFloat node.

This method will return error status if node type of target node is not IFloat type.



---

## 5.6.5. IBoolean node

### 5.6.5.1. GetBoolValue method

This method gets current value of the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetBoolValue(  
    GenApiNode node,  
    out bool value  
)
```

[VB.NET]

```
Public Function GetBoolValue (  
    node As GenApiNode,  
    <OutAttribute> ByRef value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetBoolValue(  
    GenApiNode^ node,  
    [OutAttribute] Boolean% value  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>value</i>	A variable that receives current value of the node.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

This method is available only for IBoolean node.

This method will return error status if node type of target node is not IBoolean type.

---

### 5.6.5.2. SetBoolValue method

This method sets value to the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetBoolValue(  
    GenApiNode node,  
    bool value  
)
```

[VB.NET]

```
Public Function SetBoolValue (  
    node As GenApiNode,  
    value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetBoolValue(  
    GenApiNode^ node,  
    Boolean value  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>value</i>	New value of the node.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for IBoolean node.

This method will return error status if node type of target node is not IBoolean type.

---

## 5.6.6. IEnumeration node

### 5.6.6.1. GetNumOfEnumEntries method

This method gets number of EnumEntries of the IEnumeration node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetNumOfEnumEntries(  
    GenApiNode node,  
    out int num  
)
```

[VB.NET]

```
Public Function GetNumOfEnumEntries (  
    node As GenApiNode,  
    <OutAttribute> ByRef num As Integer  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetNumOfEnumEntries(  
    GenApiNode^ node,  
    [OutAttribute] Int32% num  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>num</i>	A variable that receives number of EnumEntries of the node.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

This method is available only for IEnumeration node.

This method will return error status if node type of target node is not IEnumeration type.

The retrieved number of EnumEntries may include EnumEntries that are not valid in the camera.

---

### 5.6.6.2. GetEnumIntValue method

This method gets current value of the IEnumeration node as an integer value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetEnumIntValue(  
    GenApiNode node,  
    out long value  
)
```

[VB.NET]

```
Public Function GetEnumIntValue (  
    node As GenApiNode,  
    <OutAttribute> ByRef value As Long  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetEnumIntValue(  
    GenApiNode^ node,  
    [OutAttribute] Int64% value  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>value</i>	A variable that receives current value of the node as an integer value.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for IEnumeration node.

This method will return error status if node type of target node is not IEnumeration type.

---

### 5.6.6.3. SetEnumIntValue method

This method sets value of the IEnumeration node specifying integer value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetEnumIntValue(  
    GenApiNode node,  
    long value  
)
```

[VB.NET]

```
Public Function SetEnumIntValue (  
    node As GenApiNode,  
    value As Long  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetEnumIntValue(  
    GenApiNode^ node,  
    Int64 value  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>value</i>	New integer value of the node.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for IEnumeration node.

This method will return error status if node type of target node is not IEnumeration type.

---

#### 5.6.6.4. GetEnumStrValue method

This method gets current value of the IEnumeration node as an string value.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

[C#]

```
public CamApiStatus GetEnumStrValue(  
    GenApiNode node,  
    out string name  
)
```

[VB.NET]

```
Public Function GetEnumStrValue (  
    node As GenApiNode,  
    <OutAttribute> ByRef name As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetEnumStrValue(  
    GenApiNode^ node,  
    [OutAttribute] String^% name  
)
```

##### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>name</i>	A variable that receives current value of the node as a string value.

##### [Return value]

Returns result status.

##### [Remarks]

This method is available only for IEnumeration node.

This method will return error status if node type of target node is not IEnumeration type.

---

### 5.6.6.5. SetEnumStrValue method

This method sets value of the IEnumeration node specifying string value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetEnumStrValue(  
    GenApiNode node,  
    string name  
)
```

[VB.NET]

```
Public Function SetEnumStrValue (  
    node As GenApiNode,  
    name As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetEnumStrValue(  
    GenApiNode^ node,  
    String^ name  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>name</i>	New string value of the node.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for IEnumeration node.

This method will return error status if node type of target node is not IEnumeration type.

---

#### 5.6.6.6. GetEnumeratorByIndex method

This method gets a IEnumEntry node object specified by index in EnumEntry list of the IEnumeration node.

namespace: Teli.TeliCamAPI.NET

##### [Syntax]

---

[C#]

```
public CamApiStatus GetEnumeratorByIndex(  
    GenApiNode node,  
    int enumEntryIndex,  
    ref GenApiNode entryNode  
)
```

[VB.NET]

```
Public Function GetEnumeratorByIndex (  
    node As GenApiNode,  
    enumEntryIndex As Integer,  
    ByRef entryNode As GenApiNode  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetEnumeratorByIndex(  
    GenApiNode^ node,  
    Int32 enumEntryIndex,  
    GenApiNode^% entryNode  
)
```

##### [Parameters]

---

Parameters	Description
<i>node</i>	Parent IEnumeration node object.
<i>enumEntryIndex</i>	A variable that receives current value of the node as a string value.
<i>entryNode</i>	A variable that receives retrieved IEnumEntry node object.

##### [Return value]

---

Returns result status.

##### [Remarks]

---

This method is available only for IEnumeration node.

This method will return error status if node type of target node is not IEnumeration type.



---

## 5.6.7. IEnumEntry node

### 5.6.7.1. GetEnumEntryIntValue method

This method gets the IEnumEntry node value as an integer value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetEnumEntryIntValue(  
    GenApiNode node,  
    out long value  
)
```

[VB.NET]

```
Public Function GetEnumEntryIntValue (  
    node As GenApiNode,  
    <OutAttribute> ByRef value As Long  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetEnumEntryIntValue(  
    GenApiNode^ node,  
    [OutAttribute] Int64% value  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>value</i>	A variable that receives current value of the node as an integer value.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

This method is available only for IEnumEntry node.

This method will return error status if node type of target node is not IEnumEntry type.

---

### 5.6.7.2. GetEnumEntryStrValue method

This method gets the IEnumEntry node value as a string value.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus GetEnumEntryStrValue(  
    GenApiNode node,  
    out string name  
)
```

[VB.NET]

```
Public Function GetEnumEntryStrValue (  
    node As GenApiNode,  
    <OutAttribute> ByRef name As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetEnumEntryStrValue(  
    GenApiNode^ node,  
    [OutAttribute] String^% name  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>name</i>	A variable that receives current value of the node as a string value.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for IEnumEntry node.

This method will return error status if node type of target node is not IEnumEntry type.

---

## 5.6.8. ICommand node

### 5.6.8.1. ExecuteCommand method

This method sends command of the ICommand type node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus ExecuteCommand(  
    GenApiNode node  
)
```

[VB.NET]

```
Public Function ExecuteCommand (  
    node As GenApiNode  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus ExecuteCommand(  
    GenApiNode^ node  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for ICommand node.

This method will return error status if node type of target node is not ICommand type.

---

### 5.6.8.2. IsCommandDone method

This method gets the status whether the command has finished or not,

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus IsCommandDone(  
    GenApiNode node,  
    out bool value  
)
```

[VB.NET]

```
Public Function IsCommandDone (  
    node As GenApiNode,  
    <OutAttribute> ByRef value As Boolean  
) As CamApiStatus
```

[C++]

```
public:  
    CamApiStatus IsCommandDone(  
        GenApiNode^ node,  
        [OutAttribute] Boolean % value  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>value</i>	A variable that receives status whether the command has finished or not,.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for ICommand node.

This method will return error status if node type of target node is not ICommand type.

---

## 5.6.9. IString node

### 5.6.9.1. GetStrValue method

This method gets string value of the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public CamApiStatus GetStrValue(  
    GenApiNode node,  
    out string name  
)
```

[VB.NET]

```
Public Function GetStrValue (  
    node As GenApiNode,  
    <OutAttribute> ByRef name As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus GetStrValue(  
    GenApiNode^ node,  
    [OutAttribute] String^% name  
)
```

#### [Parameters]

---

Parameters	Description
<i>node</i>	Target node object.
<i>name</i>	A variable that receives current value of the node as a string value.

#### [Return value]

---

Returns result status.

#### [Remarks]

---

This method is available only for IString node.

This method will return error status if node type of target node is not IString type.

---

### 5.6.9.2. SetStrValue method

This method sets string value to the node.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

[C#]

```
public CamApiStatus SetStrValue(  
    GenApiNode node,  
    string name  
)
```

[VB.NET]

```
Public Function SetStrValue (  
    node As GenApiNode,  
    name As String  
) As CamApiStatus
```

[C++]

```
public:  
CamApiStatus SetStrValue(  
    GenApiNode^ node,  
    String^ name  
)
```

#### [Parameters]

Parameters	Description
<i>node</i>	Target node object.
<i>name</i>	New string value for the node.

#### [Return value]

Returns result status.

#### [Remarks]

This method is available only for IString node.

This method will return error status if node type of target node is not IString type.

---

## 5.6.10.Properties

### 5.6.10.1. Parent property

This property gets parent [CameraDevice](#) object.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public CameraDevice Parent { get; }
```

[VB.NET]

```
Public ReadOnly Property Parent As CameraDevice  
    Get
```

[C++]

```
public:  
property CameraDevice^ Parent {  
    CameraDevice^ get ();  
}
```

---

## 5.7. CamSystemInfo class

CamSystemInfo is a class for providing information of TeliCameraDNetAPI system.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

---

[C#]

```
public class CamSystemInfo
```

[VB.NET]

```
Public Class CamSystemInfo
```



[C++]

```
public ref class CamSystemInfo
```

The followings are members of CamSystemInfo class available for user applications.



### [Member Variables (Fields)]

---

	Name	Description
	gevInfo	GevSystemInfo class object for providing system information of GigE Vision API.
	u3vInfo	U3vSystemInfo class object for providing system information of USB3 Vision API.

### [Properties]

---

	Name	Description
	APIDllVersion	Gets version of TeliCamAPI.dll.
	APIDNetDllVersion	Gets version of TeliCamDNetApi.dll.



---

## 5.7.1. Member Variables (Fields)

### 5.7.1.1. gevInfo field

GevSystemInfo object for providing GigE Vision API system information.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public GevSystemInfo gevInfo
```

[VB.NET]

```
Public gevInfo As GevSystemInfo
```

[C++]

```
public:  
GevSystemInfo^ gevInfo
```

### 5.7.1.2. u3vInfo field

U3vSystemInfo object for providing USB3 Vision API system information.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public U3vSystemInfo u3vInfo
```

[VB.NET]

```
Public u3vInfo As U3vSystemInfo
```

[C++]

```
public:  
U3vSystemInfo^ u3vInfo
```

---

## 5.7.2. Properties

### 5.7.2.1. APIDllVersion property

This property gets version of TeliCamAPI.dll.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public string APIDllVersion { get; }
```

[VB.NET]

```
Public ReadOnly Property APIDllVersion As String  
    Get
```

[C++]

```
public:  
property String^ APIDllVersion {  
    String^ get ();  
}
```

### 5.7.2.2. APIDNetDllVersion property

This property gets version of TeliCamDNetApi.dll.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public string APIDNetDllVersion { get; }
```

[VB.NET]

```
Public ReadOnly Property APIDNetDllVersion As String  
    Get
```

[C++]

```
public:  
property String^ APIDNetDllVersion {  
    String^ get ();  
}
```

---

## 5.8. GevSystemInfo class

GevSystemInfo is a class for providing system information of GigE Vision API.

namespace: Teli.TeliCamAPI.NET

### **[Syntax]**

---

[C#]

```
public class GevSystemInfo
```

[VB.NET]

```
Public Class GevSystemInfo
```




[C++]

```
public ref class GevSystemInfo
```

The followings are members of GevSystemInfo class available for user applications.

### **[Properties]**

---

	Name	Description
	DllExVersion	Gets version of TeliGevCamApi.dll.
	DllVersion	Gets version of TeliGevApi2.dll.
	DriverVersion	Gets version of TeliGevDriver.sys.

---

## 5.8.1. Properties

### 5.8.1.1. DllExVersion property

This property gets version of TeliGevCamApi.dll.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public string DllExVersion { get; }
```

[VB.NET]

```
Public ReadOnly Property DllExVersion As String  
    Get
```

[C++]

```
public:  
property String^ DllExVersion {  
    String^ get ();  
}
```

### 5.8.1.2. DllVersion property

This property gets version of TeliGevApi2.dll

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public string DllVersion { get; }
```

[VB.NET]

```
Public ReadOnly Property DllVersion As String  
    Get
```

[C++]

```
public:  
property String^ DllVersion {  
    String^ get ();  
}
```

---

### 5.8.1.3. DriverVersion property

This property gets version of TeliGevDriver.sys.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public string DriverVersion { get; }
```

[VB.NET]

```
Public ReadOnly Property DriverVersion As String  
    Get
```

[C++]

```
public:  
property String^ DriverVersion {  
    String^ get ();  
}
```

---

## 5.9. U3vSystemInfo class

U3vSystemInfo is a class for providing system information of USB3 Vision API.

namespace: Teli.TeliCamAPI.NET

### **[Syntax]**

---

[C#]

```
public class U3vSystemInfo
```

[VB.NET]

```
Public Class U3vSystemInfo
```




[C++]

```
public ref class U3vSystemInfo
```

The followings are members of U3vSystemInfo class available for user applications.

### **[Properties]**

---

	Name	Description
	DllExVersion	Gets version of TeliU3vCamApi.dll.
	DllVersion	Gets version of TeliU3vApi2.dll.
	DriverVersion	Gets version of TeliU3vDriver.sys.

---

## 5.9.1. Properties

### 5.9.1.1. DllExVersion property

This property gets version of TeliU3vCamApi.dll

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public string DllExVersion { get; }
```

[VB.NET]

```
Public ReadOnly Property DllExVersion As String  
    Get
```

[C++]

```
public:  
property String^ DllExVersion {  
    String^ get ();  
}
```

### 5.9.1.2. DllVersion property

This property gets version of TeliU3vApi2.dll.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public string DllVersion { get; }
```

[VB.NET]

```
Public ReadOnly Property DllVersion As String  
    Get
```

[C++]

```
public:  
property String^ DllVersion {  
    String^ get ();  
}
```

---

### 5.9.1.3. DriverVersion property

This property gets version of TeliU3vDriver.sys.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public string DriverVersion { get; }
```

[VB.NET]

```
Public ReadOnly Property DriverVersion As String  
    Get
```

[C++]

```
public:  
property String^ DriverVersion {  
    String^ get ();  
}
```



---

## 5.10.CameraInfo class

CameraInfo is a class for providing information of the camera.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

---

[C#]

```
public class CameraInfo
```

[VB.NET]

```
Public Class CameraInfo
```



[C++]

```
public ref class CameraInfo
```

The followings are members of CameraInfo class available for user applications.






### [Member Variables (Fields)]

---

	Name	Description
	gevCamInfo	GevCameraInfo class object for providing information of GigE Vision camera.
	u3vCamInfo	U3vCameraInfo class object for providing information of USB3 Vision camera.

### [Properties]

---

	Name	Description
	CamType	Gets Interface type of the camera.
	Manufacturer	Gets manufacture name of the camera.
	ModelName	Gets model name of the camera.
	SerialNumber	Gets serial number of the camera.
	UserDefineName	Gets name of the camera defined by user.

---

## 5.10.1.Member Variables (Fields)

### 5.10.1.1. gevCamInfo field

GevCameraInfo object for providing GigE Vision camera information.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public GevCameraInfo gevCamInfo
```

[VB.NET]

```
Public gevCamInfo As GevCameraInfo
```

[C++]

```
public:  
GevCameraInfo^ gevCamInfo
```

### 5.10.1.2. u3vCamInfo field

U3vCameraInfo object for providing USB3 Vision camera information.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public U3vCameraInfo u3vCamInfo
```

[VB.NET]

```
Public u3vCamInfo As U3vCameraInfo
```

[C++]

```
public:  
U3vCameraInfo^ u3vCamInfo
```

---

## 5.10.2.Properties

### 5.10.2.1. CamType property

This property gets Interface type of the camera.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CameraType CamType { get; }
```

[VB.NET]

```
Public ReadOnly Property CamType As CameraType  
    Get
```

[C++]

```
public:  
property CameraType CamType {  
    CameraType get ();  
}
```

### 5.10.2.2. Manufacturer property

This property gets manufacture name of the camera.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public string Manufacturer { get; }
```

[VB.NET]

```
Public ReadOnly Property Manufacturer As String  
    Get
```

[C++]

```
public:  
property String^ Manufacturer {  
    String^ get ();  
}
```

---

#### 5.10.2.3. ModelName property

This property gets model name of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public string ModelName { get; }
```

[VB.NET]

```
Public ReadOnly Property ModelName As String  
    Get
```

[C++]

```
public:  
property String^ ModelName {  
    String^ get ();  
}
```

#### 5.10.2.4. SerialNumber property

This property gets serial number of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public string SerialNumber { get; }
```

[VB.NET]

```
Public ReadOnly Property SerialNumber As String  
    Get
```

[C++]

```
public:  
property String^ SerialNumber {  
    String^ get ();  
}
```

---

#### 5.10.2.5. UserDefinedName property

This property gets name of the camera defined by user.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public string UserDefinedName { get; }
```

[VB.NET]

```
Public ReadOnly Property UserDefinedName As String  
    Get
```

[C++]

```
public:  
property String^ UserDefinedName {  
    String^ get ();  
}
```

##### **[Remarks]**

---

When user application writes new user defined name, the old user defined name may remain until "[GetNumOfCameras\(\)](#)" of [CameraSystem](#) object is called.

---

## 5.11. GevCameraInfo class

GevCameraInfo is a class for providing information of a GigE Vision camera.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

---

[C#]

```
public class GevCameraInfo
```

[VB.NET]

```
Public Class GevCameraInfo
```

















[C++]

```
public ref class GevCameraInfo
```

The followings are members of GevCameraInfo class available for user applications.

### [Properties]

---

	Name	Description
	AdapterDisplayName	Gets display name of the network adapter that the camera is connected.
	AdapterGateway	Gets default gateway IP address of the network adapter that the camera is connected.
	AdapterIPAddress	Gets IP address of the network adapter that the camera is connected.
	AdapterMACAddress	Gets MAC address of the network adapter that the camera is connected.
	AdapterSubnet	Gets subnet mask of the network adapter that the camera is connected.
	CurrentIP_DHCP	Gets DHCP status (active / inactive) of the camera.
	CurrentIP_LLA	Gets LLA status (active / inactive) of the camera.
	CurrentIP_Persistent	Gets Persistent IP status (active / inactive) of the camera.
	DisplayName	Gets display name of the camera.
	Gateway	Gets default gateway IP address of the camera.
	IPAddress	Gets IP address of the camera.
	MACAddress	Gets MAC address of the camera.
	Subnet	Gets subnet mask of the camera.
	SupportIP_DHCP	Gets DHCP feature implementation status of the camera.
	SupportIP_LLA	Gets LLA feature implementation status of the camera.
	SupportIP_Persistent	Gets Persistent IP feature implementation status of the camera.

---

## 5.11.1.Properties

### 5.11.1.1. AdapterDisplayName property

This property gets display name of the network adapter that the camera is connected.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public string AdapterDisplayName { get; }
```

[VB.NET]

```
Public ReadOnly Property AdapterDisplayName As String  
    Get
```

[C++]

```
public:  
property String^ AdapterDisplayName {  
    String^ get ();  
}
```

### 5.11.1.2. AdapterGateway property

This property gets default gateway IP address of the network adapter that the camera is connected.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public byte[] AdapterGateway { get; }
```

[VB.NET]

```
Public ReadOnly Property AdapterGateway As Byte()  
    Get
```

[C++]

```
public:  
property array<Byte>^ AdapterGateway {  
    array<Byte>^ get ();  
}
```

---

#### 5.11.1.3. AdapterIPAddress property

This property gets IP address of the network adapter that the camera is connected.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public byte[] AdapterIPAddress { get; }
```

[VB.NET]

```
Public ReadOnly Property AdapterIPAddress As Byte()  
    Get
```

[C++]

```
public:  
property array<Byte>^ AdapterIPAddress {  
    array<Byte>^ get ();  
}
```

#### 5.11.1.4. AdapterMACAddress property

This property gets MAC address of the network adapter that the camera is connected.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public byte[] AdapterMACAddress { get; }
```

[VB.NET]

```
Public ReadOnly Property AdapterMACAddress As Byte()  
    Get
```

[C++]

```
public:  
property array<Byte>^ AdapterMACAddress {  
    array<Byte>^ get ();  
}
```



---

#### 5.11.1.5. AdapterSubnet property

This property gets subnet mask of the network adapter that the camera is connected.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public byte[] AdapterSubnet { get; }
```

[VB.NET]

```
Public ReadOnly Property AdapterSubnet As Byte()  
    Get
```

[C++]

```
public:  
property array<Byte>^ AdapterSubnet {  
    array<Byte>^ get ();  
}
```

#### 5.11.1.6. CurrentIP\_DHCP property

This property gets DHCP status (active / inactive) of the camera.  
If true, DHCP feature is active.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public bool CurrentIP_DHCP { get; }
```

[VB.NET]

```
Public ReadOnly Property CurrentIP_DHCP As Boolean  
    Get
```

[C++]

```
public:  
property bool CurrentIP_DHCP {  
    bool get ();  
}
```

---

#### 5.11.1.7. CurrentIP\_LLA property

This property gets LLA (Link Local Address) status (active / inactive) of the camera.  
if true, LLA is active.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public bool CurrentIP_LLA { get; }
```

[VB.NET]

```
Public ReadOnly Property CurrentIP_LLA As Boolean  
    Get
```

[C++]

```
public:  
property bool CurrentIP_LLA {  
    bool get ();  
}
```

#### 5.11.1.8. CurrentIP\_Persistent property

This property gets Persistent IP status (active / inactive) of the camera.  
If true, Persistent IP is active.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public bool CurrentIP_Persistent { get; }
```

[VB.NET]

```
Public ReadOnly Property CurrentIP_Persistent As Boolean  
    Get
```

[C++]

```
public:  
property bool CurrentIP_Persistent {  
    bool get ();  
}
```

---

#### 5.11.1.9. DisplayName property

This property gets display name of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public string DisplayName { get; }
```

[VB.NET]

```
Public ReadOnly Property DisplayName As String  
    Get
```

[C++]

```
public:  
property String^ DisplayName {  
    String^ get ();  
}
```

#### 5.11.1.10. Gateway property

This property gets default gateway IP address of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public byte[] Gateway { get; }
```

[VB.NET]

```
Public ReadOnly Property Gateway As Byte()  
    Get
```

[C++]

```
public:  
property array<Byte>^ Gateway {  
    array<Byte>^ get ();  
}
```

---

#### 5.11.1.11. IPAddressproperty

This property gets IP address of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public byte[] IPAddress { get; }
```

[VB.NET]

```
Public ReadOnly Property IPAddress As Byte()  
    Get
```

[C++]

```
public:  
property array<Byte>^ IPAddress {  
    array<Byte>^ get ();  
}
```

#### 5.11.1.12. MACAddress property

This property gets MAC address of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public byte[] MACAddress { get; }
```

[VB.NET]

```
Public ReadOnly Property MACAddress As Byte()  
    Get
```

[C++]

```
public:  
property array<Byte>^ MACAddress {  
    array<Byte>^ get ();  
}
```

---

#### 5.11.1.13. Subnet property

This property gets subnet mask of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public byte[] Subnet { get; }
```

[VB.NET]

```
Public ReadOnly Property Subnet As Byte()  
    Get
```

[C++]

```
public:  
property array<Byte>^ Subnet {  
    array<Byte>^ get ();  
}
```

#### 5.11.1.14. SupportIP\_DHCP property

This DHCP feature implementation status of the camera.

If true, DHCP feature is implemented.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public bool SupportIP_DHCP { get; }
```

[VB.NET]

```
Public ReadOnly Property SupportIP_DHCP As Boolean  
    Get
```

[C++]

```
public:  
property bool SupportIP_DHCP {  
    bool get ();  
}
```

---

#### 5.11.1.15. SupportIP\_LLA property

This property gets LLA feature implementation status of the camera.  
If true, LLA feature is implemented.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public bool SupportIP_LLA { get; }
```

[VB.NET]

```
Public ReadOnly Property SupportIP_LLA As Boolean  
    Get
```

[C++]

```
public:  
property bool SupportIP_LLA {  
    bool get ();  
}
```

#### 5.11.1.16. SupportIP\_Persistent property

This property gets Persistent IP feature implementation status of the camera.  
If true, persistent IP feature is implemented.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public bool SupportIP_Persistent { get; }
```

[VB.NET]

```
Public ReadOnly Property SupportIP_Persistent As Boolean  
    Get
```

[C++]

```
public:  
property bool SupportIP_Persistent {  
    bool get ();  
}
```

---

## 5.12. U3vCameraInfo class

U3vCameraInfo is a class for providing information of USB3 Vision camera.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

---

[C#]

```
public class U3vCameraInfo
```

[VB.NET]

```
Public Class U3vCameraInfo
```







[C++]

```
public ref class U3vCameraInfo
```

The followings are members of U3vCameraInfo class available for user applications.

### [Properties]

---

	Name	Description
	AdapterDeviceID	Gets device ID of USB chip on the USB3.0 adapter that the camera is connected.
	AdapterDfltMaxPacketSize	Gets default MaxPacketSize value of the USB3.0 adapter that the camera is connected.
	AdapterVendorID	Gets vender ID of USB chip on the USB3.0 adapter that the camera is connected.
	DeviceVersion	Gets device version of the camera.
	FamilyName	Gets family name of the camera.
	ManufacturerInfo	Gets information of manufacturer of the camera.

---

## 5.12.1.Properties

### 5.12.1.1. AdapterDeviceID property

This property gets device ID of USB chip on the USB3.0 adapter that the camera is connected.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public uint AdapterDeviceID { get; }
```

[VB.NET]

```
Public ReadOnly Property AdapterDeviceID As UInteger  
    Get
```

[C++]

```
public:  
property UInt32 AdapterDeviceID {  
    UInt32 get ();  
}
```

### 5.12.1.2. AdapterDfltMaxPacketSize property

This property gets default MaxPacketSize value of the USB3.0 adapter that the camera is connected.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public int AdapterDfltMaxPacketSize { get; }
```

[VB.NET]

```
Public ReadOnly Property AdapterDfltMaxPacketSize As Integer  
    Get
```

[C++]

```
public:  
property Int32 AdapterDfltMaxPacketSize {  
    Int32 get ();  
}
```

---

**[Remarks]**

The default MaxPacketSize value is used when 0 was specified to “maxPacketSize” property on calling “[Open\(\)](#)” method of [CameraStream](#) class.

The default MaxPacketSize value varies, depending on vender of USB chip. Vender can be identified by venderID.



---

#### 5.12.1.3. AdapterVendorID property

This property gets vender ID of USB chip on the USB3.0 adapter that the camera is connected.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public uint AdapterVendorID { get; }
```

[VB.NET]

```
Public ReadOnly Property AdapterVendorID As UInteger  
    Get
```

[C++]

```
public:  
property UInt32 AdapterVendorID {  
    UInt32 get ();  
}
```

#### 5.12.1.4. DeviceVersion property

This property gets device version of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public string DeviceVersion { get; }
```

[VB.NET]

```
Public ReadOnly Property DeviceVersion As String  
    Get
```

[C++]

```
public:  
property String^ DeviceVersion {  
    String^ get ();  
}
```

---

#### 5.12.1.5. FamilyName property

This property gets family name of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public string FamilyName { get; }
```

[VB.NET]

```
Public ReadOnly Property FamilyName As String  
    Get
```

[C++]

```
public:  
property String^ FamilyName {  
    String^ get ();  
}
```

#### 5.12.1.6. ManufacturerInfo property

This property gets information of manufacturer of the camera.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public string ManufacturerInfo { get; }
```

[VB.NET]

```
Public ReadOnly Property ManufacturerInfo As String  
    Get
```

[C++]

```
public:  
property String^ ManufacturerInfo {  
    String^ get ();  
}
```

---

## 5.13. CameraImageInfo class

CameraImageInfo is a class for providing information of an image received from the camera.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

---

[C#]

```
public class CameraImageInfo
```

[VB.NET]

```
Public Class CameraImageInfo
```













[C++]

```
public ref class CameraImageInfo
```

The followings are members of CameraImageInfo class available for user applications.

### [Properties]

---

	Name	Description
	Timestamp	Gets Timestamp of the image that the camera sent.
	PixelFormat	Gets PixelFormat of the image that the camera sent.
	SizeX	Gets horizontal size of the image that the camera sent, in pixels.
	SizeY	Gets vertical size of the image that the camera sent, in pixels.
	OffsetX	Gets horizontal offset of the image that the camera sent, in pixels.
	OffsetY	Gets vertical offset of the image that the camera sent, in pixels.
	PaddingX	Gets horizontal padding of the image that the camera sent, in bytes.
	BlockID	Gets frame index of the image data that the camera assigned.
	BufferPointer	Gets pointer to the image data.
	Size	Gets size of the image data in bytes.
	ImageID	Gets image index of the image that TeliCamDNetAPI (TeliCamAPI) assigned.
	Status	Gets result status of the image on reception of it.

---

## 5.13.1.Properties

### 5.13.1.1. Timestamp property

This property gets Timestamp of the image that the camera sent.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public ulong Timestamp { get; }
```

[VB.NET]

```
Public ReadOnly Property Timestamp As ULong  
    Get
```

[C++]

```
public:  
property UInt64 Timestamp {  
    UInt64 get ();  
}
```

---

**[Remarks]**

The unit of Timestamp value is 1 nano second in USB3 Vision Camera, 16 nano seconds in GigE Vision Camera.

### 5.13.1.2. PixelFormat property

This property gets PixelFormat of the image that the camera sent.

namespace: Teli.TeliCamAPI.NET

---

**[Syntax]**

[C#]

```
public CameraPixelFormat PixelFormat { get; }
```

[VB.NET]

```
Public ReadOnly Property PixelFormat As CameraPixelFormat  
    Get
```

[C++]

```
public:  
property CameraPixelFormat PixelFormat {  
    CameraPixelFormat get ();  
}
```

---

#### 5.13.1.3. SizeX property

This property gets horizontal size of the image that the camera sent, in pixels.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public int SizeX { get; }
```

[VB.NET]

```
Public ReadOnly Property SizeX As Integer  
    Get
```

[C++]

```
public:  
property Int32 SizeX {  
    Int32 get ();  
}
```

#### 5.13.1.4. SizeY property

This property gets vertical size of the image that the camera sent, in pixels.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public int SizeY { get; }
```

[VB.NET]

```
Public ReadOnly Property SizeY As Integer  
    Get
```

[C++]

```
public:  
property Int32 SizeY {  
    Int32 get ();  
}
```

---

#### 5.13.1.5. OffsetX property

This property gets horizontal offset of the image that the camera sent, in pixels.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public int OffsetX { get; }
```

[VB.NET]

```
Public ReadOnly Property OffsetX As Integer  
    Get
```

[C++]

```
public:  
property Int32 OffsetX {  
    Int32 get ();  
}
```

#### 5.13.1.6. OffsetY property

This property gets vertical offset of the image that the camera sent, in pixels.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public int OffsetY { get; }
```

[VB.NET]

```
Public ReadOnly Property OffsetY As Integer  
    Get
```

[C++]

```
public:  
property Int32 OffsetY {  
    Int32 get ();  
}
```

---

#### 5.13.1.7. PaddingX property

This property gets horizontal padding of the image that the camera sent, in bytes.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public int PaddingX { get; }
```

[VB.NET]

```
Public ReadOnly Property PaddingX As Integer  
    Get
```

[C++]

```
public:  
property Int32 PaddingX {  
    Int32 get ();  
}
```

#### 5.13.1.8. BlockID property

This property gets frame index of the image data that the camera assigned.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

---

[C#]

```
public ulong BlockID { get; }
```

[VB.NET]

```
Public ReadOnly Property BlockID As ULong  
    Get
```

[C++]

```
public:  
property UInt64 BlockID {  
    UInt64 get ();  
}
```

##### **[Remarks]**

---

BlockID will be incremented on every streaming out of an image.

BlockID will be cleared to 0 when image acquisition is stopped.

---

#### 5.13.1.9. BufferPointer property

This property gets pointer to the image data.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

[C#]

```
public ulong BlockID { get; }
```

[VB.NET]

```
Public ReadOnly Property BlockID As ULong  
    Get
```

[C++]

```
public:  
property UInt64 BlockID {  
    UInt64 get ();  
}
```

##### **[Remarks]**

The image data buffers in ImageRingBuffer are under the management of TeliCamDNetAPI (TeliCamAPI).

#### 5.13.1.10. Size property

This property gets size of the image data in bytes.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

[C#]

```
public int Size { get; }
```

[VB.NET]

```
Public ReadOnly Property Size As Integer  
    Get
```

[C++]

```
public:  
property Int32 Size {  
    Int32 get ();  
}
```



---

#### 5.13.1.11. ImageID property

This property gets image index of the image that TeliCamDNetAPI (TeliCamAPI) assigned.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

[C#]

```
public ulong ImageID { get; }
```

[VB.NET]

```
Public ReadOnly Property ImageID As ULong  
    Get
```

[C++]

```
public:  
property UInt64 ImageID {  
    UInt64 get ();  
}
```

##### **[Remarks]**

Image index will be incremented on every reception of an image regardless of result status, success or failure.

#### 5.13.1.12. Status property

This property gets result status of the image on reception of it.

namespace: Teli.TeliCamAPI.NET

##### **[Syntax]**

[C#]

```
public CamApiStatus Status { get; }
```

[VB.NET]

```
Public ReadOnly Property Status As CamApiStatus  
    Get
```

[C++]

```
public:  
property CamApiStatus Status {  
    CamApiStatus get ();  
}
```

---

## 5.14. GenApiNode class

GenApiNode is a Node class of camera description data, used in [GenApiWrapper](#) class

namespace: Teli.TeliCamAPI.NET

### **[Syntax]**

---

[C#]

```
public class GenApiNode
```

[VB.NET]

```
Public Class GenApiNode
```

[C++]

```
public ref class GenApiNode
```

---

## 5.15. ImageAcquiredEventArgs class

ImageAcquiredEventArgs is an event argument class for [ImageAcquired](#) event.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

---

[C#]

```
public class ImageAcquiredEventArgs : EventArgs
```

[VB.NET]

```
Public Class ImageAcquiredEventArgs  
    Inherits EventArgs
```



[C++]

```
public ref class ImageAcquiredEventArgs : public EventArgs
```

The followings are members of ImageAcquiredEventArgs class available for user applications.

### [Properties]

---

	Name	Description
	BufferIndex	Gets index of StreamRequest in ImageRingBuffer that contains the received image data.
	ImageInfo	Gets CameraImageInfo object that contains information of the received image.

---

## 5.15.1.Properties

### 5.15.1.1. BufferIndex property

This property gets index of StreamRequest in ImageRingBuffer that contains the received image data.

namespace: Teli.TeliCamAPI.NET

---

#### [Syntax]

[C#]

```
public int BufferIndex { get; }
```

[VB.NET]

```
Public ReadOnly Property BufferIndex As Integer  
    Get
```

[C++]

```
public:  
property Int32 BufferIndex {  
    Int32 get ();  
}
```

### 5.15.1.2. ImageInfo property

This property gets CameraImageInfo object that contains information of the received image.

namespace: Teli.TeliCamAPI.NET

---

#### [Syntax]

[C#]

```
public CameraImageInfo ImageInfo { get; }
```

[VB.NET]

```
Public ReadOnly Property ImageInfo As CameraImageInfo  
    Get
```

[C++]

```
public:  
property CameraImageInfo^ ImageInfo {  
    CameraImageInfo^ get ();  
}
```

---

## 5.16. ImageErrorReceivedEventArgs class

ImageErrorReceivedEventArgs is an event argument class for [ImageErrorReceived](#) event.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

---

[C#]

```
public class ImageErrorReceivedEventArgs : EventArgs
```

[VB.NET]

```
Public Class ImageErrorReceivedEventArgs  
    Inherits EventArgs
```



[C++]

```
public ref class ImageErrorReceivedEventArgs : public EventArgs
```

The followings are members of ImageErrorReceivedEventArgs class available for user applications.

### [Properties]

---

	Name	Description
	BufferIndex	Gets index of StreamRequest in ImageRingBuffer that contains the received image data.
	ErrorStatus	Gets status code for the error.

---

## 5.16.1.Properties

### 5.16.1.1. BufferIndex property

This property gets index of StreamRequest in ImageRingBuffer that contains the received image data.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public int BufferIndex { get; }
```

[VB.NET]

```
Public ReadOnly Property BufferIndex As Integer  
    Get
```

[C++]

```
public:  
property Int32 BufferIndex {  
    Int32 get ();  
}
```

### 5.16.1.2. ErrorStatus property

This property gets status code for the error.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public CamApiStatus ErrorStatus { get; }
```

[VB.NET]

```
Public ReadOnly Property ErrorStatus As CamApiStatus  
    Get
```

[C++]

```
public:  
property CamApiStatus ErrorStatus {  
    CamApiStatus get ();  
}
```

---

## 5.17. BufferBusyReceivedEventArgs class

BufferBusyReceivedEventArgs is an event argument class for [BufferBusyReceived](#) event.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

---

[C#]

```
public class BufferBusyReceivedEventArgs : EventArgs
```

[VB.NET]

```
Public Class BufferBusyReceivedEventArgs  
    Inherits EventArgs
```


[C++]

```
public ref class BufferBusyReceivedEventArgs : public EventArgs
```

The followings are members of BufferBusyReceivedEventArgs class available for user applications.

### [Properties]

---

	Name	Description
	BufferIndex	Gets index of StreamRequest in ImageRingBuffer that CameraStream object failed to save received image data.

---

## 5.17.1.Properties

### 5.17.1.1. BufferIndex property

This property gets index of StreamRequest in ImageRingBuffer that CameraStream object failed to save received image data.

namespace: Teli.TeliCamAPI.NET

#### **[Syntax]**

---

[C#]

```
public int BufferIndex { get; }
```

[VB.NET]

```
Public ReadOnly Property BufferIndex As Integer  
    Get
```

[C++]

```
public:  
property Int32 BufferIndex {  
    Int32 get ();  
}
```



---

## 5.18. CameraUtility class

CameraUtility is a class for providing image handling utility methods.

All members in this class is static member. User application can use utility methods in this class without creating instance of CameraUtility class.

namespace: Teli.TeliCamAPI.NET

### [Syntax]

[C#]

```
public class CameraUtility
```

[VB.NET]












```
Public Class CameraUtility
```

[C++]

```
public ref class CameraUtility
```

The followings are members of CameraUtility class available for user applications.

### [Methods]

	Name	Description
Image format converter		
	PrepareLUT	Prepares look up table for image format converter,
	Convert*ToBGRA	Converts raw image data to BGRA 32bit format data.
	Convert*ToBGR	Converts raw image data to BGR 24bit format data.
	ConvertImage	Converts various types of source image data to BGR or BGRA format data, specifying source and destination pixel format.
Miscellaneous utilities		
	GetBitPerPixel	Gets bit per pixel data of the argument PixelFormat.
	GetDataDepth	Gets data depth of the argument PixelFormat.
	IsMonochromic	Gets bit per pixel data of the argument PixelFormat.
	IsPixelBayer	Gets whether the argument PixelFormat is monochromic.
	SaveBmp*	Saves specified image data as a Bitmap file.
	ReverseImage	Creates an image reversed horizontally and / or vertically.
	CopyMemoryT	Copies a block of unmanaged memory from one location to another.

---

## 5.18.1. Image format converter

### 5.18.1.1. PrepareLUT method

This method Prepares look up table for image format converter,

namespace: Teli.TeliCamAPI.NET.Utility

---

**[Syntax]**

[C#]

```
public static CamApiStatus PrepareLUT()
```

[VB.NET]

```
Public Shared Function PrepareLUT As CamApiStatus
```

[C++]

```
public:  
static CamApiStatus PrepareLUT()
```

---

**[Return value]**

Returns result status.

---

**[Remarks]**

User application should call this function once, before calling image format converter.

---

### 5.18.1.2. Convert\*ToBGRA methods

CameraUtility class provides converters for converting 23 kinds of source image data to BGRA format data.

BGRA format is the format that data of a pixel consists of four 8bit components (B: blue, G: green, R: red, A: alpha (transparency)). B data is located at the smallest address and A data is located at the largest address. Data layout of BGRA format is same as image data portion of 32bit ARGB format Bitmap object.

Value 0xFF is set to transparency component (A) in these methods.

namespace: Teli.TeliCamAPI.NET.Utility

#### [Syntax]

---

[C#]

```
public static CamApiStatus Convert*PToBGRA(  
    IntPtr dstBGRA,  
    IntPtr src,  
    int width,  
    int height  
)
```

[VB.NET]

```
Public Shared Function Convert*PToBGRA (  
    dstBGRA As IntPtr,  
    src As IntPtr,  
    width As Integer,  
    height As Integer  
) As CamApiStatus
```

[C++]

```
public:  
static CamApiStatus Convert*PToBGRA (  
    IntPtr dstBGRA,  
    IntPtr src,  
    Int32 width,  
    Int32 height  
)
```

#### [Methods]

---

Methods	Source PixelFormat	PixelFormat ID
ConvertMono8ToBGRA	Mono8	0x01080001
ConvertMono10ToBGRA	Mono10	0x01100003
ConvertMono12ToBGRA	Mono12	0x01100005
ConvertMono16ToBGRA	Mono16	0x01100007
ConvertByrGR8ToBGRA	BayerGR8	0x01080008
ConvertByrRG8ToBGRA	BayerRG8	0x01080009
ConvertByrGB8ToBGRA	BayerGB8	0x0108000A
ConvertByrBG8ToBGRA	BayerBG8	0x0108000B

Methods	Source PixelFormat	PixelFormat ID
ConvertByrGR10ToBGRA	BayerGR10	0x0110000C
ConvertByrRG10ToBGRA	BayerRG10	0x0110000D
ConvertByrGB10ToBGRA	BayerGB10	0x0110000E
ConvertByrBG10ToBGRA	BayerBG10	0x0110000F
ConvertByrGR12ToBGRA	BayerGR12	0x01100010
ConvertByrRG12ToBGRA	BayerRG12	0x01100011
ConvertByrGB12ToBGRA	BayerGB12	0x01100012
ConvertByrBG12ToBGRA	BayerBG12	0x01100013
ConvertRGB8PToBGRA	RGB8 (RGB8Packed)	0x02180014
ConvertBGR8PToBGRA	BGR8 (BGR8Packed)	0x02180015
ConvertBGR10PToBGRA	BGR10 (BGR10Packed)	0x02300019
ConvertBGR12PToBGRA	BGR12 (BGR12Packed)	0x0230001B
ConvertYUV411PToBGRA	YUV411_8_UYYVYY ((YUV411Packed)	0x020C001E
ConvertYUV422PToBGRA	YUV422_8_UYVY (YUV422Packed)	0x0210001F
ConvertYUV444PToBGRA	YUV8_UYV (YUV444Packed)	0x02180020

#### [Parameters]

Parameters	Description
<i>dstBGRA</i>	IntPtr to destination image data for receiving converted BGRA data. Memory should be allocated beforehand.
<i>src</i>	IntPtr to source image data.
<i>width</i>	Width of source image data in pixels. Width must be multiple of 4.
<i>height</i>	Height of source image data in pixels.

#### [Return value]

Returns result status.

---

### 5.18.1.3. Convert\*ToBGR method

CameraUtility class provides converters for converting 23 kinds of source image data to BGR format data.

BGR format is the format that data of a pixel consists of three 8bit components (B: blue, G: green, R: red). B data is located at the smallest address and R data is located at the largest address. Data layout of BGR format is same as image data portion of 24bit RGB format Bitmap object.

namespace: Teli.TeliCamAPI.NET.Utility

#### [Syntax]

---

[C#]

```
public static CamApiStatus Convert*PToBGR(  
    IntPtr dstBGR,  
    IntPtr src,  
    int width,  
    int height  
)
```

[VB.NET]

```
Public Shared Function Convert*PToBGR (  
    dstBGR As IntPtr,  
    src As IntPtr,  
    width As Integer,  
    height As Integer  
) As CamApiStatus
```

[C++]

```
public:  
static CamApiStatus Convert*PToBGR (  
    IntPtr dstBGR,  
    IntPtr src,  
    Int32 width,  
    Int32 height  
)
```

#### [Methods]

---

Methods	Source PixelFormat	PixelFormat ID
ConvertMono8ToBGR	Mono8	0x01080001
ConvertMono10ToBGR	Mono10	0x01100003
ConvertMono12ToBGR	Mono12	0x01100005
ConvertMono16ToBGR	Mono16	0x01100007
ConvertByrGR8ToBGR	BayerGR8	0x01080008
ConvertByrRG8ToBGR	BayerRG8	0x01080009
ConvertByrGB8ToBGR	BayerGB8	0x0108000A
ConvertByrBG8ToBGR	BayerBG8	0x0108000B
ConvertByrGR10ToBGR	BayerGR10	0x0110000C
ConvertByrRG10ToBGR	BayerRG10	0x0110000D

Methods	Source PixelFormat	PixelFormat ID
ConvertByrGB10ToBGR	BayerGB10	0x0110000E
ConvertByrBG10ToBGR	BayerBG10	0x0110000F
ConvertByrGR12ToBGR	BayerGR12	0x01100010
ConvertByrRG12ToBGR	BayerRG12	0x01100011
ConvertByrGB12ToBGR	BayerGB12	0x01100012
ConvertByrBG12ToBGR	BayerBG12	0x01100013
ConvertRGB8PToBGR	RGB8 (RGB8Packed)	0x02180014
ConvertBGR8PToBGR	BGR8 (BGR8Packed)	0x02180015
ConvertBGR10PToBGR	BGR10 (BGR10Packed)	0x02300019
ConvertBGR12PToBGR	BGR12 (BGR12Packed)	0x0230001B
ConvertYUV411PToBGR	YUV411_8_UYYVYY (YUV411Packed)	0x020C001E
ConvertYUV422PToBGR	YUV422_8_UYVY (YUV422Packed)	0x0210001F
ConvertYUV444PToBGR	YUV8_UYV (YUV444Packed)	0x02180020

#### [Parameters]

Parameters	Description
<i>dstBGR</i>	IntPtr to destination image data for receiving converted BGRA data. Memory should be allocated beforehand.
<i>src</i>	IntPtr to source image data.
<i>width</i>	Width of source image data in pixels. Width must be multiple of 4.
<i>height</i>	Height of source image data in pixels.

#### [Return value]

Returns result status.

---

#### 5.18.1.4. ConvertImage

This method converts various type of source image data to BGR or BGRA format data, using methods in section 5.18.1.2 Convert\*ToBGRA methods and 5.18.1.3 Convert\*ToBGR method.

namespace: Teli.TeliCamAPI.NET.Utility

##### [Syntax]

---

[C#]

```
public static CamApiStatus ConvertImage(  
    DstPixelFormat dstPixelFormat,  
    CameraPixelFormat srcPixelFormat,  
    bool bayerConversion,  
    IntPtr dst,  
    IntPtr src,  
    int width,  
    int height  
)
```

[VB.NET]

```
Public Shared Function ConvertImage (  
    dstPixelFormat As DstPixelFormat,  
    srcPixelFormat As CameraPixelFormat,  
    bayerConversion As Boolean,  
    dst As IntPtr,  
    src As IntPtr,  
    width As Integer,  
    height As Integer  
) As CamApiStatus
```

[C++]

```
public:  
static CamApiStatus ConvertImage(  
    DstPixelFormat dstPixelFormat,  
    CameraPixelFormat srcPixelFormat,  
    Boolean bayerConversion,  
    IntPtr dst,  
    IntPtr src,  
    Int32 width,  
    Int32 height  
)
```

##### [Parameters]

---

Parameters	Description
<i>dstPixelFormat</i>	<a href="#">Pixel format of destination image data.</a>
<i>srcPixelFormat</i>	<a href="#">Pixel format of source image data.</a>
<i>bayerConversion</i>	Flag whether convert Bayer type source image regarding color filter. If source PixelFormat is not Bayer type, this parameter is ignored.
<i>dst</i>	IntPtr to destination image data that receives converted image data. Memory should be allocated beforehand.

---

Parameters	Description
<i>src</i>	IntPtr to source image data.
<i>width</i>	Width of source image data in pixels. Width must be multiple of 4.
<i>height</i>	Height of source image data in pixels.

**[Return value]**

Returns result status.



---

## 5.18.2. Miscellaneous utilities

### 5.18.2.1. GetBitPerPixel method

This method returns bit per pixel data of the argument `PixelFormat`.

When pixel data consists of multiple components, summation of all components bit count will be returned.

namespace: `Teli.TeliCamAPI.NET.Utility`

#### [Syntax]

[C#]

```
public static byte GetBitPerPixel(  
    CameraPixelFormat pixelFormat  
)
```

[VB.NET]

```
Public Shared Function GetBitPerPixel (  
    pixelFormat As CameraPixelFormat  
) As Byte
```

[C++]

```
public:  
static Byte GetBitPerPixel(  
    CameraPixelFormat pixelFormat  
)
```

#### [Parameters]

Parameters	Description
<i>pixelFormat</i>	<code>PixelFormat</code>

#### [Return value]

Returns bit per pixel data of the argument `PixelFormat`.

For example, 24 will be returned when *pixelFormat* is 'RGB8'.

---

### 5.18.2.2. GetDataDepth method

This method returns data depth of the argument `PixelFormat`, in bits.

When pixel data consists of multiple components, depth of a component will be returned.

namespace: `Teli.TeliCamAPI.NET.Utility`

#### [Syntax]

---

[C#]

```
public static byte GetDataDepth(  
    CameraPixelFormat pixelFormat  
)
```

[VB.NET]

```
Public Shared Function GetDataDepth (  
    pixelFormat As CameraPixelFormat  
) As Byte
```

[C++]

```
public:  
static Byte GetDataDepth(  
    CameraPixelFormat pixelFormat  
)
```

#### [Parameters]

---

Parameters	Description
<i>pixelFormat</i>	<code>PixelFormat</code>

#### [Return value]

---

Returns data depth in bits.

For example, 8 will be returned when *pixelFormat* is 'RGB8'.

---

### 5.18.2.3. IsMonochromic method

This method returns whether the argument `PixelFormat` is monochromic or not.

Monochromic format is a format that the most significant byte of `PixelFormat` is "01".

Mono8, Mono10, Mono12, Mono16, and Bayer formats are Monochromic format.

namespace: Teli.TeliCamAPI.NET.Utility

#### [Syntax]

---

[C#]

```
public static bool IsMonochromic(  
    CameraPixelFormat pixelFormat  
)
```

[VB.NET]

```
Public Shared Function IsMonochromic (  
    pixelFormat As CameraPixelFormat  
) As Boolean
```

[C++]

```
public:  
static Boolean IsMonochromic(  
    CameraPixelFormat pixelFormat  
)
```

#### [Parameters]

---

Parameters	Description
<i>pixelFormat</i>	PixelFormat

#### [Return value]

---

Returns true when *pixelFormat* is Mono8, Mono10, Mono12, Mono16, or Bayer formats.  
Otherwise, returns false.

---

#### 5.18.2.4. IsPixelBayer methods

This method returns whether the argument *PixelFormat* is Bayer type format or not.

namespace: Teli.TeliCamAPI.NET.Utility

##### [Syntax]

---

[C#]

```
public static bool IsPixelBayer(  
    CameraPixelFormat pixelFormat  
)
```

[VB.NET]

```
Public Shared Function IsPixelBayer (  
    pixelFormat As CameraPixelFormat  
) As Boolean
```

[C++]

```
public:  
static Boolean IsPixelBayer(  
    CameraPixelFormat pixelFormat  
)
```

##### [Parameters]

---

Parameters	Description
<i>pixelFormat</i>	PixelFormat

##### [Return value]

---

Returns true if *pixelFormat* is Bayer type format, otherwise returns false.

---

### 5.18.2.5. SaveBmp\* method

This method saves argument image data as a Bitmap file.

Three methods are available, saving as 32bit ARGB Bitmap, 24bit RGB Bitmap, or 8bit indexed Bitmap (for monochrome image).

If argument *path* file already exists, this method will overwrite the existing file with new Bitmap file.

namespace: Teli.TeliCamAPI.NET.Utility

#### [Syntax]

[C#]

```
public static CamApiStatus SaveBmp*(  
    IntPtr target,  
    int width,  
    int height,  
    string path  
)
```

[VB.NET]

```
Public Shared Function SaveBmp* (  
    target As IntPtr,  
    width As Integer,  
    height As Integer,  
    path As String  
) As CamApiStatus
```

[C++]

```
public:  
static CamApiStatus SaveBmp*(  
    IntPtr target,  
    Int32 width,  
    Int32 height,  
    String^ path  
)
```

#### [Methods]

Methods	Bitmap Pixel Format	Remarks
SaveBmpARGB	Format32bppArgb	
SaveBmpRGB	Format24bppRgb	
SaveBmpMono	Format8bppIndexed	Uses color pallet for monochrome image.

#### [Parameters]

Parameters	Description
<i>target</i>	IntPtr to target image data.
<i>width</i>	Width of target image in pixels.
<i>height</i>	Height of target image in pixels.

---

Parameters	Description
<i>path</i>	File path for saving the created bitmap. Folder of the oath should be accessible as writable folder.

**[Return value]**

Returns result status.

---

### 5.18.2.6. ReverseImage method

This method creates image reversed horizontally and / or vertically.

When source image is Bayer type, image data will be reversed as monochrome image data, which means that the color filter layout of the reversed image will be different from that of source image.

For example, when BayerBG image is horizontally reversed, BayerGB image will be created as result image. When BayerBG image is vertically reversed, BayerGR image will be created as result image.

This method can handle the following PixelFormat images.

CameraPixelFormat.Mono8,	CameraPixelFormat.Mono10,	CameraPixelFormat.Mono12
CameraPixelFormat.BayerGR8,	CameraPixelFormat.BayerRG8,	
CameraPixelFormat.BayerGB8,	CameraPixelFormat.BayerBG8,	
CameraPixelFormat.BayerGR10,	CameraPixelFormat.BayerRG10,	
CameraPixelFormat.BayerGB10,	CameraPixelFormat.BayerBG10,	
CameraPixelFormat.YUV411_8,	CameraPixelFormat.YUV422_8,	
CameraPixelFormat.RGB8,	CameraPixelFormat.BGR8	

namespace: Teli.TeliCamAPI.NET.Utility

#### **[Syntax]**

[C#]

```
public static CamApiStatus ReverseImage(  
    IntPtr dst,  
    IntPtr src,  
    CameraPixelFormat pixelFormat,  
    int width,  
    int height,  
    bool reverseX,  
    bool reverseY  
)
```

[VB.NET]

```
Public Shared Function ReverseImage (  
    dst As IntPtr,  
    src As IntPtr,  
    pixelFormat As CameraPixelFormat,  
    width As Integer,  
    height As Integer,  
    reverseX As Boolean,  
    reverseY As Boolean  
) As CamApiStatus
```

---

```

[C++]
public:
    static CamApiStatus ReverseImage(
        IntPtr dst,
        IntPtr src,
        CameraPixelFormat pixelFormat,
        Int32 width,
        Int32 height,
        Boolean reverseX,
        Boolean reverseY
    )

```

#### **[Parameters]**

---

Parameters	Description
<i>dst</i>	IntPtr to destination image data that receives reversed image.
<i>src</i>	IntPtr to source image data.
<i>pixelFormat</i>	PixelFormat of source image.
<i>width</i>	Width of source image in pixels.
<i>height</i>	Height of source images in pixels.
<i>reverseX</i>	Flag wheter reverse forizontally or not. If true, horizontally reversed image will be created.
<i>reverseY</i>	Flag wheter reverse Vertically or not. If true, vertically reversed image will be created.

#### **[Return value]**

---

Returns result status.



---

### 5.18.2.7. CopyMemoryT method

This method copies a block of unmanaged memory from one location to another.

namespace: Teli.TeliCamAPI.NET.Utility

#### **[Syntax]**

---

[C#]

```
public static void CopyMemoryT(  
    IntPtr destination,  
    IntPtr source,  
    int length  
)
```

[VB.NET]

```
Public Shared Sub CopyMemoryT (  
    destination As IntPtr,  
    source As IntPtr,  
    length As Integer  
)
```

[C++]

```
public:  
static void CopyMemoryT(  
    IntPtr destination,  
    IntPtr source,  
    Int32 length  
)
```

#### **[Parameters]**

---

Parameters	Description
<i>destination</i>	IntPtr to destination memory block.
<i>source</i>	IntPtr to source memory block.
<i>length</i>	Length of copying data in bytes.

#### **[Return value]**

---

This method does not have return value.

#### **[Remarks]**

---

This method is a wrapper of CopyMemory() function of Win32 API.






If the source and destination blocks overlap, the results are undefined.

---

## 6. EventHandler delegates

TeliCamDNetAPI provides the following EventHandler delegates for user applications.

### [Delegates]

	Name	Description
	RemovedEventHandler	Represents the method that will handle a <a href="#">Removed</a> event of <a href="#">CameraDevice</a> class.
	ImageAcquiredEventHandler	Represents the method that will handle an <a href="#">ImageAcquired</a> event of <a href="#">CameraStream</a> class.
	ImageErrorReceivedEventHandler	Represents the method that will handle an <a href="#">ImageErrorReceived</a> event of <a href="#">CameraStream</a> class.
	BufferBusyReceivedEventHandler	Represents the method that will handle a <a href="#">BufferBusyReceived</a> event of <a href="#">CameraStream</a> class.
	CameraEventReceivedEventHandler	Represents the method that will handle a CameraEvents of <a href="#">CameraEvent</a> class.

---

### 6.1.1. RemovedEventHandler delegate

This EventHandler delegate represents the method that will handle a [Removed](#) event of [CameraDevice](#) class.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public delegate void RemovedEventHandler(  
    CameraDevice sender,  
    EventArgs e  
)
```

[VB.NET]

```
Public Delegate Sub RemovedEventHandler (  
    sender As CameraDevice,  
    e As EventArgs  
)
```

[C++]

```
public delegate void RemovedEventHandler(  
    CameraDevice^ sender,  
    EventArgs^ e  
)
```

#### [Parameters]

---

Parameters	Description
<i>sender</i>	The source of the event.
<i>e</i>	An object that contains the event data. EventArgs.Empty will be returned.

#### [Remarks]

---

To associate the event with the method that will handle the event, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.

---

### 6.1.2. ImageAcquiredEventHandler delegate

This EventHandler delegate represents the method that will handle an [ImageAcquired](#) event of [CameraStream](#) class.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public delegate void ImageAcquiredEventHandler(  
    CameraStream sender,  
    ImageAcquiredEventArgs e  
)
```

[VB.NET]

```
Public Delegate Sub ImageAcquiredEventHandler (  
    sender As CameraStream,  
    e As ImageAcquiredEventArgs  
)
```

[C++]

```
public delegate void ImageAcquiredEventHandler(  
    CameraStream^ sender,  
    ImageAcquiredEventArgs^ e  
)
```

#### [Parameters]

---

Parameters	Description
<i>sender</i>	The source of the event.
<i>e</i>	An object that contains the event data. Information of the received image and index of StreamRequest that contains the received image are contained in the event data.

#### [Remarks]

---

To associate the event with the method that will handle the event, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.

---

### 6.1.3. ImageErrorReceivedEventHandler delegate

This EventHandler delegate represents the method that will handle an [ImageErrorReceived](#) event of [CameraStream](#) class.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public delegate void ImageErrorReceivedEventHandler(  
    CameraStream sender,  
    ImageErrorReceivedEventArgs e  
)
```

[VB.NET]

```
Public Delegate Sub ImageErrorReceivedEventHandler (  
    sender As CameraStream,  
    e As ImageErrorReceivedEventArgs  
)
```

[C++]

```
public delegate void ImageErrorReceivedEventHandler(  
    CameraStream^ sender,  
    ImageErrorReceivedEventArgs^ e  
)
```

#### [Parameters]

---

Parameters	Description
<i>sender</i>	The source of the event.
<i>e</i>	An object that contains the event data. Status code of the error and index of StreamRequest that contains the received image are contained in the event data.

#### [Remarks]

---

To associate the event with the method that will handle the event, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.

---

### 6.1.4. BufferBusyReceivedEventHandler delegate

This EventHandler delegate represents the method that will handle a [BufferBusyReceived](#) event of [CameraStream](#) class.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public delegate void BufferBusyReceivedEventHandler (
    CameraStream sender,
    BufferBusyReceivedEventArgs e
)
```

[VB.NET]

```
Public Delegate Sub BufferBusyReceivedEventHandler (
    sender As CameraStream,
    e As BufferBusyReceivedEventArgs
)
```

[C++]

```
public delegate void BufferBusyReceivedEventHandler(
    CameraStream^ sender,
    BufferBusyReceivedEventArgs^ e
)
```

#### [Parameters]

---

Parameters	Description
<i>sender</i>	The source of the event.
<i>e</i>	An object that contains the event data. Index of StreamRequest that <a href="#">CameraStream</a> object failed to save the received image is contained in the event data.

#### [Remarks]

---

To associate the event with the method that will handle the event, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.

---

### 6.1.5. CameraEventReceivedEventHandler delegate

This EventHandler delegate represents the method that will handle a CameraEvents of [CameraEvent](#) class.

namespace: Teli.TeliCamAPI.NET

#### [Syntax]

---

[C#]

```
public delegate void CameraEventReceivedEventHandler(  
    CameraEvent sender,  
    EventArgs e  
)
```

[VB.NET]

```
Public Delegate Sub CameraEventReceivedEventHandler (  
    sender As CameraEvent,  
    e As EventArgs  
)
```

[C++]

```
public delegate void CameraEventReceivedEventHandler(  
    CameraEvent^ sender,  
    EventArgs^ e  
)
```

#### [Parameters]

---

Parameters	Description
<i>sender</i>	The source of the event.
<i>e</i>	An object that contains the event data. EventArgs.Empty will be returned.

#### [Remarks]

---

To associate the event with the method that will handle the event, add an instance of the delegate to the event. The event handler is called whenever the event occurs, unless you remove the delegate.

---

## 7. Enumeration types

TeliCamDNetAPI provides the following Enumeration types for user applications.

### [Enumerations]

---

namespace: Teli.TeliCamAPI.NET

Name	Description
CamApiStatus	Status code.
CameraType	Interface type of camera.
CameraAccessMode	Access mode (Control Channel Privilege) of camera.
CameraAcquisitionMode	Mode of capturing and transferring image.
CameraEventType	CameraEvent (message).
NodeType	Type of camera description data node.
NodeAccessMode	Access mode of a node.
NodeVisibility	Visibility of a node (desirable user level for accessing a node).
NodeCachingMode	Caching mode of a node.
NodeRepresentation	Representation of node value.
NodeFloatNotation	Notation of float node value.
CameraImageFormat	Image format of camera
CameraPixelFormat	PixelFormat of image stream.
CameraTestPattern	Test pattern choice.
CameraAcqFrameRateCtrl	Frame rate control mode.
CameraImageBufferMode	Status of image buffer in the camra.
CameraTriggerSequence	Exposure time controlling sequence.
CameraTriggerSource	Trigger source of random trigger shutter.
CameraExposureTimeCtrl	Control mode of exposure time.
CameraLineSelector	Selctor for I/O line in the camera.
CameraLineSource	Source signal of I/O line in the camera.
CameraTimerTriggerSource	Trigger source signal of Timer0Active sigal.
CameraGainAuto	Status of AGC.
CameraBalanceRatioSelector	Color component gain selector for White Balance.
CameraBalanceWhiteAuto	Mode of autonmatic white balance control.
CameraColorCorrectionMatrixSelector	Element selector for color correction matrix.
CameraUserSetSelector	Channel selector for UserSet feature.

namespace: Teli.TeliCamAPI.NET

Name	Description
DstPixelFormat	Pixel data format for output image of image format converter.



---

## 7.1. CamApiStatus

CamApiStatus type specified status code

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Success	It Succeeded
NotInitialize	The initialization for API has never been performed.
AlreadyInitialized	The initialization for API has already been performed.
NotFound	API detected no cameras.
AlreadyOpened	Camera or the other object has been already opened,
AlreadyActivated	Camera or the other object has been already activated,
InvalidCameraIndex	The specified camera index is invalid.
InvalidCameraHandle	The specified camera handle is invalid.
InvalidNodeHandle	The specified node handle is invalid.
InvalidStreamHandle	The specified stream handle is invalid.
InvalidRequestHandle	The specified request handle is invalid.
InvalidEventHandle	The specified event handle is invalid.
InvalidParameter	The specified parameter is invalid.
BufferTooSmall	The specified buffer is too small.
NoMemory	To allocate internal buffer of API is unsuccessful
MemoryNoAccess	To access to the specified buffer is unsuccessful.
NotImplemented	Feature is not implemented in the camera or API. This status may be also returned when wrong register name or node name is used,
Timeout	The timeout occurred.
CameraNotResponding	The specified camera does not send response. The cable which has connected the camra may have separated. Please check the connection,
EmptyCompleteQueue	The request in the Complete Queue is empty.
NotReady	The target is not ready state.
AccessModeSetError	Api failed to set access mode of the camera.
IoDeviceError	Controller caused an I/O error.
XML_LoadError	Api failed to load XML file (camera description file).
GenICamError	Error occurred in GenApi dll.
DLL_LoadError	Api failed to load DLL file.
InvalidAddress	The specified address is invalid.
WriteProtect	The register is protected from writing data.
BadAlignment	The address is not aligned to specified boundary.
AccessDenied	Accessing data was denied. User application may not have access privilege.
Busy	The camera is in busy state. Try again after a while.
NotReadable	Timeout occurred in requesting stream or event.
NotWritable	The StreamRequests could not be completed in time after the first packet was received. (only in GigE Vision camera)
NotAvailable	The next packet could not be received in time after somepacket was received. (Only in GigE Vision camera)

Member name	Description
RequestTimeout	The received data size exceeded maximum size specified.
ResendTimeout	The data size actually received was different from the sizedescribed in the trailer.
ResponseTimeout	The received packet count exceeded the maximum.
BufferFull	The packet returned error status.
UnexpectedBufferSize	Resend command is not implemented in the camera.
UnexpectedNumber	The packet is unavailable.
PacketStatusError	A leader of the next block has been received before the completion of a frame data. Packets may be lost.
ResendNotImplemented	The requestwas flushed by the user.
PacketUnavailable	The loss pf packet exceeded the specified value (default:20) In GigE Vision camera case, band width of the network may not be enough for sending images in current parameters. Enabling Jumbo-Packet or restricting frame rate will be required.
MissingPackets	The request was flushed due to a change of the power state.
FlushRequested	The request was flushed due to a disconnection wth the camera.
TooManyPacketMissing	Api failed to load Driver.
FlushedByD0Exit	Api failed to open file,
FlushedByCameraRemove	Api failed to warite data in file
Driver_LoadError	Api failed to read data from file.
FileOpenError	The specified file was not found.
FileWriteError	Other error occurred.
FileReadError	It Succeeded
FileNotFound	The initialization for API has never been performed.
DataDiscarded	The initialization for API has already been performed.
InvalidParameterFromCam	Invalid parameter error returned from camera.
PayloadSizeNotAligned	The value written to the SI streaming size registers is not aligned to Payload Size Alignment value of the SI Info register.
Unsuccessful	API detected no cameras.

## 7.2. CameraType

CamerType specifies interface type of the camera

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
TypeAll	All interfaces.
TypeGev	GigE Vision interface
TypeU3v	USB3 Vision interface
Unknown	Interfaces that TeliCamDNetAPI cannot handle.

---

## 7.3. CameraAccessmode

CameraAccessMode specifies access mode (Control Channel Privilege) for GigE Vision camera.

namespace: Teli.TeliCamAPI.NET

### **[Members]**

Member name	Description
Exclusive	User application can fully control the camera. The other application cannot open the camera.
Control	User application can fully control the camera. The other application can read registers of the camera, cannot write data.
Open	User application can read registers of the camera, cannot write data.

## 7.4. CameraAcquisitionMode

CameraAcquisitionMode specifies mode of capturing and transferring image.

namespace: Teli.TeliCamAPI.NET

### **[Members]**

Member name	Description
Continuous	Repeats capturing and transferring image data continuously, until Acquisition Stop command is sent.
MultiFrame	Repeats capturing and transferring image data until a number of frames specified by AcquisitionFrameCount register has been transferred.
ImageBufferRead	Transfers image data from Image Buffer in the camera.

---

## 7.5. CameraEventType

CameraEventType specifies type of CameraEvent (message).

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
FrameTrigger	Accepted trigger signal for acquiring an image.
FrameTriggerError	Received trigger signal for acquiring image at invalid timing.
FrameTriggerWait	Started accepting trigger signal for image acquisition.
FrameTransferStart	Started transferring image stream of a frame,
FrameTransferEnd	Completed transferring image stream of a frame,
ExposureStart	Started exposure for a frame.
ExposureEnd	Finished exposure for a frame.
Timer0Start	Timer0 started counting.
Timer0End	Timer0 finished counting.
ALCLatestInformation	Updated Automatic Luminance Control data.
ALCConverged	Converged Automatic Luminance Control operation.

## 7.6. NodeType

NodeType specifies type of camera description data node.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Value	Value node. (IValue interface)
Base	Base node. (IBase interface)
Integer	Integer node. (IInteger interface)
Boolean	Boolean node. (IBoolean interface)
Command	Command node. (ICommand interface)
Float	Float node. (IFloat interface)
String	String node. (IString interface)
Register	Register node. (IRegister interface)
Category	Category node. (ICategory interface)
Enumeration	Enumeration node. (IEnumeration interface)
EnumEntry	EnumEntry node. (IEnumEntry interface)
Port	Port node (IPort node)
Unknown	Unknown.

---

## 7.7. NodeAccessMode

NodeAccessmode specifies access mode of a node.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
NotImplemented	Not implemented.
NotAvailable	Not available
WriteOnly	Write only.
ReadOnly	Read only.
ReadAndWrite	Read and Write available.
Undefined	Node object is not initialized,
Unknown	Unknown.

## 7.8. NodeVisibility

NodeVisibility specifies visibility (desirable user level for accessing a node) of a node.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Beginner	All users can accesss the node safely.
Expert	Experts and Guru can access the node safely.
Guru	Well trained specialist can access the node safely.
Invisible	No users can access the node.
Undefined	Node object is not initialized,
Unknown	Unknown.

## 7.9. NodeCachingMode

NodeCachingMode specifies caching mode of a node.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
NoCache	Not use cache.
WriteThrough	A value written to the camera is written to the cache as well.
WriteAround	Only read values are written to the cache.
Undefined	Node object is not initialized,
Unknown	Unknown.

---

## 7.10. NodeRepresentation

NodeRepresentation specifies Representation of node value

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Linear	Slider with linear behavior.
Logarithmic	Slider with logarithmic behavior.
Boolean	Check box.
PureNumber	Decimal number in an edit control.
HexNumber	Hex number in an edit control.
IPv4Address	IP address,
MACAddress	MAC address.
Undefined	Node object is not initialized,
Unknown	Unknown.

## 7.11. NodeFloatNotationn

NodeFloatNotation specifies Notation of float node value

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Automatic	The notation if either scientific or fixed depending on what is shorter.
Fixed	The notation is fixed, e.g. 123.4
Scientific	The notation is scientific, e.g. 1.234e2
Undefined	Object is not yet initialized
Unknown	Unknown.

## 7.12. CameraImageFormat

CameraImageFormat specifies image format of the camera.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Format0	Format0 (Scalable Mode)
Fomat1	Format1 (Binning Mode)
Format2	Format2 (Decimation Mode)

---

## 7.13. CameraPixelFormat

CameraPixelFormat specifies PixelFormat of image stream.

namespace: Teli.TeliCamAPI.NET

### **[Members]**

Member name	Description
Mono8	
Mono10	
Mono12	
Mono16	
BayerGR8	
BayerGR10	
BayerGR12	
BayerRG8	
BayerRG10	
BayerRG12	
BayerGB8	
BayerGB10	
BayerGB12	
BayerBG8	
BayerBG10	
BayerBG12	
RGB8	
BGR8	
BGR10	
BGR12	
YUV411_8	
YUV422_8	
YUV8	

---

## 7.14. CameraTestPattern

CameraTestPattern specifies Test patter (test image) of the camera.

namespace: Teli.TeliCamAPI.NET

### **[Members]**

Member name	Description
Off	
Black	
White	
GreyA	
GreyB	
HotizontalRamp	
GreyScale	
ColorBar	
VertiacI Ramp	

## 7.15. CameraAcqFrameRateCtrl

CameraAcqFrameRateCtrl specifies control mode of frame rate.

namespace: Teli.TeliCamAPI.NET

### **[Members]**

Member name	Description
NoSpecify	Settings other than AcquisitionFrameRate (for example, Exposure Time, etc.,) has priority.
Manual	AcquisitionFrameRate parameter has priority over the other parameters.

## 7.16. CameraImageBufferMode

CameraImageBuffer specifies status of image buffer in the camra.

namespace: Teli.TeliCamAPI.NET

### **[Members]**

Member name	Description
Off	Image buffer mode is disabled.
On	Image buffer mode is enabled.



---

## 7.17. CameraTriggerSequence

CameraTriggerSequence specifies exposure time controlling sequence.

namespace: Teli.TeliCamAPI.NET

### **[Members]**

Member name	Description
Sequence0	TriggerSequence0 (External edge mode) USB3 Vision Camera : TriggerSequence register = TriggerSequence0  GigE Vision Camera : ExposureMode register = Timed TriggerSelector register = FrameStart
Sequence1	TriggerSequence1 (External level mode) USB3 Vision Camera : TriggerSequence register = TriggerSequence1  GigE Vision Camera : ExposureMode register = TriggerWidth TriggerSelector register = FrameStart
Sequence6	TriggerSequence6 (Bulk / FrameBurst trigger mode) USB3 Vision Camera : TriggerSequence register = TriggerSequence6  GigE Vision Camera : ExposureMode register = Timed TriggerSelector register = FrameBurstStart

## 7.18. CameraTriggerSource

CameraTriggerSource specifies trigger source signal of random trigger shutter.

namespace: Teli.TeliCamAPI.NET

### **[Members]**

Member name	Description
Line0	Line0 (Hardware trigger)
Line1	Line1
Line2	Line2
Software	Software trigger.

---

## 7.19. CameraExposureTimeCtrl

CameraExposureTimeCtrl specifies control mode of exposure time.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
NoSpecify	Settings other than exposure time (for example, AcquisitionFrameRate, etc.,) has priority. USB3 Vision Camera : ExposureTimeControl register = NoSpecify  GigE Vision Camera : Not Available
Manual	ExposureTime register value has priority over the other parameters. USB3 Vision Camera : ExposureTimeControl register = Manual  GigE Vision Camera : ExposureAuto register = Off
Auto	Automatic exposure time control mode. USB3 Vision Camera : ExposureTimeControl register = Auto  GigE Vision Camera : ExposureAuto register = Continuous

## 7.20. CameraLineSelector

CameraLineSelector specifies selection of I/O line in the camera.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Line0	Line0
Line1	Line1
Line2	Line2

---

## 7.21. CameraLineSource

CameraLineSource specifies source signal for I/O line in the camera.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Off	Not use general purpose output port.
VD	VD sync. signal.
UserOutput	Value specified using UserOutputValueAll method.
Timer0Active	Timer0 active signal. Delay time and duration parameters are available.
AcquisitionActive	Signal indicating that image acquisition is active.
FrameTriggerWait	Signal indicating that the camera can accept trigger signal, for random trigger shutter mode,
FrameActive	Signal indicating that exposure and transfer operation is being done
FrameTransferActive	Signal indicating that image is being transferred on the streaming interface.
ExposureActive	Signal indicating that exposure operation is active.

## 7.22. CameraTimerTriggerSource

CameraTimerTriggerSource specifies trigger source signal for Timer0.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Off	Not use Timer.
Line0Active	Starts timer count when Line0 (TRIG_IN) signal becomes active.
FrameTrigger	Starts timer count when trigger signal for image acquisition is accepted.
ExposureStart	Starts timer count when exposure starts,
ExposureEnd	Starts timer count when exposure finished,

## 7.23. CameraGainAuto

CameraGainAuto specifies AGC status of the camera.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Off	Manual mode.
Auto	Automatic Gain Control is active.

---

## 7.24. CameraBalanceRatioSelector

CameraBalanceRatioSelector specifies target color component for setting White balance.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Blue	Balance Ratio (B Gain)
Red	Balance Ratio (R Gain)

## 7.25. CameraBalanceWhiteAuto

CameraBalanceWhiteAuto specifies mode of automatic white balance.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Off	No operation.
Continuous	Execute auto white balance continuously.
Once	Execute auto white balance once.

## 7.26. CameraColorCorrectionMatrixSelector

CameraColorCorrectionMatrixSelector specifies element of color correction matrix.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
RG	SelectorI = R, SelectorJ = G.
RB	SelectorI = R, SelectorJ = B.
GR	SelectorI = G, SelectorJ = R.
GB	SelectorI = G, SelectorJ = B.
BR	SelectorI = B, SelectorJ = R.
BG	SelectorI = B, SelectorJ = G.

---

## 7.27. CameraUserSetSelector

CameraUserSetSelector specifies channel of UserSet memory.

namespace: Teli.TeliCamAPI.NET

### [Members]

Member name	Description
Default	Initial factory parameter values. This channel is read-only.
UserSet1	UserSet channel 1
UserSet2	UserSet channel 2
UserSet3	UserSet channel 3
UserSet4	UserSet channel 4
UserSet5	UserSet channel 5
UserSet6	UserSet channel 6
UserSet7	UserSet channel 7
UserSet8	UserSet channel 8
UserSet9	UserSet channel 9
UserSet10	UserSet channel 10
UserSet11	UserSet channel 11
UserSet12	UserSet channel 12
UserSet13	UserSet channel 13
UserSet14	UserSet channel 14
UserSet15	UserSet channel 15

## 7.28. DstPixelFormat

DstPixelFormat specifies pixel data format for output image of image format converter.

namespace: Teli.TeliCamAPI.NET.Utility

### [Members]

Member name	Description
BGRA32	BGRA 32bit format.
BGR24	BGR 24bit format

---

## 8. Others

### 8.1. Revision History

Date	Version	Description
2015/02/25	1.0.0	Initial version.
2015/06/12	1.0.1	Added the Windows 8.1 description
2016/06/28	1.0.2	<ul style="list-style-type: none"><li>• Supports FrameBurst function.</li><li>• Appended function (SetLineModeAll())</li></ul>
2016/07/12	2.0.0	<ul style="list-style-type: none"><li>• Added the Windows 10 description</li><li>• Deleted the Windows XP &amp; Windows Vista description</li><li>• Supports FrameBurst function.</li><li>• Appended function (SetLineModeAll())</li></ul>
2016/12/16	2.0.1	<ul style="list-style-type: none"><li>• Added methods related to Chunk feature</li><li>• Added functions related to UserSetControl</li><li>• Added enumerators for CameraTriggerSource enumeration</li></ul>
2017/06/13	2.0.2	<ul style="list-style-type: none"><li>• Added Abort() method to Camera Stream class.</li><li>• Added a member of CameraBalanceWhiteAuto enumeration</li><li>• Added some members of CamApiStatus enumeration</li></ul>
2017/09/05	2.0.3	<ul style="list-style-type: none"><li>• Modified setting range of argument apiBufferCount of Open() in CameraStream class. (Minimum value : 3 → 1 , Maximum value : 30 → 128 )</li></ul>

### 8.2. Disclaimer

The disclaimer of this Software is described in another "License Agreement TeliCamSDK Eng.pdf".

Make sure to read this Agreement carefully before using it.

Refer to TeliCamSDK installation folder/Documents/License folder

---

## 8.3. License

Microsoft, Windows, Windows XP, Windows Vista, Windows 7, Windows 8.1, Windows 10, .NET Framework and Visual C++ are the trademark or the registered trademark of Microsoft Corporation.

GigE Vision™ and USB3 Vision™ are camera interface standard defined by AIA (Automated Imaging Association).

GenICam™ is a trademark of EMVA (European Machine Vision association).

Furthermore, the trade name used in this document is the trademark or the registered trademark of each company.

TeliCamSDK and Viewer use the library that the third party retains the copyright. Refer to TeliCamSDK installation folder/Documents/License folder for license.

## 8.4. Inquiry

If you need help with TeliCamSDK, GigE Vision camera, USB3 Vision camera, please visit the following website:

<https://secure.toshiba-teli.co.jp/ttfa/web/faq/top.html>

If you still can not solve the problem, please contact the our customer support below :

Imaging Solution Engineering section

Imaging & Communication Solution Engineering Division

4-7-1, Asahigaoka, Hino,

Tokyo 191-0065, Japan

Mail : TELI-EXT-technical-support@toshiba-teli.co.jp