



ACTi SDK-10000
ActiveX Control Edition
V1.2.93

API Reference Guide



www.acti.com

ACTi SDK-10000

This document is copyrighted, 2003 - 2012 by ACTi Corporation. All rights are reserved. ACTi Corporation reserves the right to make improvements to the products described in this manual at any time without notice.

No part of this manual may be reproduced, copied, translated or transmitted in any form or by any means without the prior written permission of the original manufacturer. Information provided in this manual is intended to be accurate and reliable. However, the original manufacturer assumes no responsibility for its use, or for any infringements upon the rights of third parties that may result from its use.

All other product names or trademarks are properties of their respective owners.

V1.2 Edition Aug, 2012

Table of Contents

1	INTRODUCTION	1-1
	BENEFITS OF STREAMING CONTROL	1-1
	STREAMING OBJECTS	1-1
	STREAMING CONTROL ARCHITECTURE	1-2
2	CONTROL PROPERTY	2-1
	MEDIA PROPERTY	2-1
	AutoReconnect	2-3
	AutoReconnectInterval	2-4
	ASEControlPort	2-5
	ASEMediaSource	2-6
	ASEMediaUserName	2-8
	ASEMediaPassword	2-10
	ASEStreamingPort	2-12
	BeginTime	2-13
	BitRate	2-14
	Brightness	2-16
	CodecType	2-18
	ContentStatus	2-20
	Contrast	2-21
	ControlPort	2-23
	DeviceChipType	2-24
	Duration	2-25
	EnableASE	2-26
	EnableBorder	2-27
	EndTime	2-28
	Fps	2-29
	FrameRateMode	2-31
	HttpPort	2-32
	Hue	2-33
	ID	2-34
	LastRecFileSize	2-35
	longLastRecFileSize	2-36
	MediaChannel	2-37
	MediaPassword	2-38
	MediaSource	2-39
	MediaType	2-40
	MediaUsername	2-42
	MulticastIP	2-43
	Mute	2-44
	MulticastPort	2-45
	NetworkStatus	2-46
	PreRecordTime	2-47
	PostRecordingTime	2-48
	RecordType	2-49
	RegisterPort	2-50
	ReplaceTimeCodeByLocalTime	2-51
	Resolution	2-52
	RTSPPort	2-54
	Saturation	2-55

StreamingPort	2-57
StreamType	2-58
strLastRecFileSize	2-59
VariableFPS	2-60
Version	2-61
Volume	2-62
DeviceType	2-63
QuadDeviceMode	2-64
TCPVideoStreamID	2-65
RTPVideoTrackNumber	2-66
RTPAudioTrackNumber	2-67
ConnectTimeout	2-68
UI PROPERTY	2-69
Caption	2-70
StretchToFit	2-71
PTZ PROPERTY	2-72
AddressID	2-73
BaudRate	2-74
PTZPanSpeed	2-75
Parity	2-76
Protocol	2-77
PTZFile	2-78
PTZStatus	2-79
PTZTiltSpeed	2-80
Vendor	2-81
MOTION DETECTION PROPERTY	2-82
MotionDetectionAlertDuration	2-83
DIO PROPERTY	2-84
DigitalInputInterval	2-85
3 CONTROL METHODS	3-1
MEDIA METHODS	3-1
StartAudioTransfer()	3-3
StopAudioTransfer()	3-4
Connect()	3-5
DecodeFrame()	3-6
DecodeFrameEx()	3-7
Disconnect()	3-8
EnableMouseMoveEvent()	3-9
FreeAudioToken()	3-10
GetAudioToken()	3-11
GetBeginTimeString()	3-12
GetEndTimeString()	3-13
Pause()	3-14
Play()	3-15
PlayByFrame()	3-16
SnapShot()	3-17
SendAudio()	3-19
SendURLCmdToSE()	3-20
SetCurrentTime()	3-21
SetDecodeI()	3-22
SetPlayDirection()	3-23
SetPlayRate()	3-24
StartAlarmRecord()	3-25
StartDecodeMode ()	3-26
StartRecord()	3-27

StartStream()	3-29
Stop()	3-30
StopAlarmRecord()	3-31
StopDecodeMode()	3-32
StopRecord()	3-33
StopPlayingAudioFile()	3-34
StopStream()	3-35
SetVideoTransformConfig()	3-36
UI METHODS	3-37
DisplayTitleBar()	3-38
EnableFullScreen()	3-39
SetControlActive()	3-40
SetDefaultBackgroundColor ()	3-41
SetTitleBarTextLayout()	3-42
SetTitleActiveColor()	3-44
SetTitleNonActiveColor()	3-45
SetTitleEventColor()	3-46
SetPlayingBackgroundColor()	3-47
SetBorderWidth()	3-48
SetBorderColor()	3-49
PTZ METHODS	3-50
DigitalZoomIn()	3-51
DigitalZoomOut()	3-52
DigitalMoveLeft ()	3-53
DigitalMoveRight ()	3-54
DigitalMoveUp ()	3-55
DigitalMoveDown ()	3-56
DigitalZoomOutMax ()	3-57
DisableMouseDigitalPTZ()	3-58
DisableMousePTZ()	3-59
DisablePTZ()	3-60
EnableMousePTZ()	3-61
EnableMouseDigitalPTZ()	3-62
EnablePTZ()	3-63
EnumerateVendor	3-64
EnumerateProtocol()	3-65
PTZBLC()	3-66
PTZDayNight()	3-67
PTZFocus()	3-68
PTZIris()	3-69
PTZMove()	3-70
PTZOSD()	3-72
PTZPreset()	3-73
PTZZoom()	3-74
SendPTZCmd()	3-75
MOTION DETECTION METHODS	3-76
DisableMotionDetection()	3-77
EnableMotionDetection()	3-78
GetMotionSensitive()	3-79
GetMotionStartX()	3-80
GetMotionStartY()	3-81
GetMotionEndX()	3-82
GetMotionEndY()	3-83
GetMotionRegionEnable()	3-84
SetMotionRegionEnable()	3-85

	SetMotionSetting()	3-86
	SetQuadMotionDetection()	3-88
	SetMotionSensitivity()	3-89
	StartMDSetup()	3-90
	StopMDSetup()	3-91
	SyncMDInfoFromDevice()	3-92
	DIO METHODS.....	3-93
	DisableDigitalInput()	3-94
	EnableDigitalInput()	3-95
	DigitalOutput	3-96
	OTHER METHODS	3-97
4	CONTROL EVENTS	4-1
	UI EVENTS	4-1
	OnConnect()	4-2
	OnConnectionRecovery()	4-3
	OnDisconnect()	4-4
	OnMouseKeyDoubleClick()	4-5
	OnMouseKeyDown()	4-6
	OnMouseKeyUp()	4-7
	OnMouseMoveNotify()	4-8
	OnNetworkLoss()	4-9
	OnPlay()	4-10
	OnPlaybackComplete()	4-11
	OnSaveImage()	4-12
	OnStartRecord()	4-13
	OnStop()	4-14
	OnStopRecord()	4-15
	OnTimeCode()	4-16
	OnVideoLoss()	4-17
	OnVideoRecovery()	4-18
	OnKeyDown()	4-19
	OnKeyUp()	4-20
	OnMDEventStart()	4-22
	OnMDEventEnd()	4-23
	OnMDSetting()	4-24
	OnMDSettingChange()	4-26
	DIO EVENTS.....	4-27
	OnDIEvent()	4-28
	OnDIEventEnd()	4-29
	OnRS232Event()	4-30
5	NEW METHOD	5-1
	NEW METHOD.....	5-1
	SetTextOut()	5-44
	MirrorImage()	5-66
	FlipImage()	5-77
	EnablePrivacyMask()	5-8
	SetPrivacyMask()	5-9
	EnableAbsPosition()	5-10
	DisableAbsPosition()	5-11
	AddMultiplePlaybackFile()	5-12
	RemoveAllMultiplePlaybackFile()	5-13
	PlayAudioFileToDevice()	5-14
	GetNumberOfMonitors()	5-15
	ShowRealFPS()	5-16
	SetRealFpsFontColor()	5-17

SetTextOutEx()	5-188
ExecuteURLCommand ()	5-2020
SendURLCmd ()	5-211
SendPTZCommand ()	5-222
RequestsToCheckDeviceChipType()	5-233
EnableFullScreenEx()	5-244
TxRS232Data()	5-255
AsyncDisconnect()	5-266
AsyncSendURLCmd()	5-277
EnableOnNewImageEvent()	5-288
SetGlobalMotionEnable()	5-299
GetGlobalMotionEnable()	5-3030
SnapshotWithTitleBar ()	5-311
GetFullScreenStatus()	5-322
CloseFullScreenWindow()	5-333
GetMouseDigitalPTZStatus()	5-344
GetPlayRate()	5-355
GetPlayDirection()	5-366
SetUtcCurrentTime ()	5-377
SetTextOutToRegion()	5-388
StartRecordEx()	5-4040
StopRecordEx()	5-411
SnapShotEx()	5-422
GetCurrentTimeByInteger()	5-444
SetCurrentTimeByInteger()	5-455
ConvertUTCtoString()	5-466
EnablePIRMotionDetection()	5-477
DisablePIRMotionDetection()	5-488
GetPIRConfig()	5-499
SetPIRConfig()	5-5050
SetMotionTime()	5-511
EnableMotionSetting()	5-522
SetMotionSettingEx()	5-533
GetMotionInfoFromDevice()	5-544
StartStreamEx()	5-555
SetDigitalOutput()	5-566
DevicePrivacyMask()	5-577
SnapShotEx4CMS()	5-588
GetCurrentImage()	5-6060
GetCurrentImageEx()	5-611
EnableDecoder()	5-622
EnableMouseMoveEvent()	5-633
SendRS232Command()	5-644
RefreshControl()	5-655
EnableTitleEvent()	5-666
DisableTitleEvent()	5-677
ExecuteURLCommandEx()	5-68
SetMotionThreshold()	5-69
DigitalPause()	5-70
DigitalPlay()	5-71
IsLeftButtonUp()	5-72
EnableCurrentImageBuffer()	5-73
DisableCurrentImageBuffer()	5-74
6 NEW PROPERTY	6-1
NEW PROPERTY	6-1

	MotionDetectionInterval	6-2
	RenderInterface	6-3
	PlayFileName	6-4
	RealFPS	6-5
	RecordLedStatus	6-6
	RecordStatus	6-7
	SiteID	6-8
	PIRDetectionAlertDuration	6-9
	DIDefaultValue	6-10
	Is4VGAMode	6-11
	NVRID	6-12
	NVRName	6-13
7	NEW EVENTS	7-1
	NEW EVENTS.....	7-1
	OnPlayAudioFileComplete()	7-2
	OnMouseMove()	7-3
	OnAsyncURLCommandDone()	7-4
	OnTimeZoneEx()	7-5
	OnTimeCodeEx()	7-6
	OnMDSettingEx()	7-7
	OnPIREventStart()	7-9
	OnPIREventEnd()	7-10
	RequestSetIOnly()	7-11
	OnDIEventEx()	7-12
	OnNewImage()	7-13
	OnAutoReconnect()	7-14

1

Introduction

Streaming Control is developed using Microsoft ATL (Active Template Library) v3.0 frameworks. By applying ATL, Streaming Control has better performance with smaller executable size.

Benefits of Streaming Control

The benefit of streaming control is listed as follow.

- **Object-oriented design:** With streaming control, you may benefit from pure object-oriented programming. There are six major objects in streaming control, they are:
- **Multiple programming Supports:** Streaming control is developed standard ActiveX Control framework, hence, it can be programmed with Microsoft Visual C++, Microsoft Visual Basic, and ASP with VBScript and HTML with Java Script.
- **Encapsulate the complexity of streaming architecture:** With streaming control, user may skips the trivial details of streaming architecture, such as TCP, UDP, RTP, Multicast, etc; instead, with simple methods, users may focus on developing business rules for applications.
- **Integrates and adopts multiple media formats:** streaming control adopts streaming and file media formats.

Streaming Objects

Streaming control consists of six major objects, they are listed as follow:

<i>Name</i>	<i>Description</i>
DIO	Controls digital input and digital output.
Media	Media manipulation. Media object can be streaming media or file media.
MotionDetection	Motion detection object handles all necessary operations when an event is triggered.
Network	Network object controls the streaming parameters, with TCP, UDP or Multicast protocols.
PTZ	PTZ object handles all operations on pan, tilt and zoom events.
UI	UI object handles the interaction between the streaming object and the user's application.

Streaming Control Architecture

Streaming control can be programmed using Visual BASIC, Visual C++, Visual Studio.Net and HTML + JavaScript, and the calling conventions are the same through various developing environment.

By adopting DirectX platform, Streaming Control is able to render standard MPEG-4 streaming file and supports other video formats in the future.

2

Control Property

This section lists media property. These values have to be defined before invoking Control methods.

Media Property

<i>Name</i>	<i>Description</i>
AutoReconnect	[I/O] Sets/Gets whether certain streaming media reconnect automatically when connection lost
AutoReconnectInterval	[I/O] Sets/Gets reconnect interval whether certain streaming media reconnect automatically when connection loss
ASEControlPort	[I/O] Sets/Gets the special control port for streaming engine
ASEMediaSource	[I/O] Identifies the source of the media stream for streaming engine
錯誤! 找不到參照來源。	[I/O] Sets the user name of the media stream for streaming engine
ASEMediaPassword	[I] Sets the password to access the media stream for streaming engine
ASEStreamingPort	[I/O] Sets/Gets the streaming port for streaming port
BeginTime	[O] The time stamp in the playback media
BitRate	[I/O] Sets/Gets the bit-rate value of video configuration
Brightness	[I/O] Sets/Gets the brightness value of video configuration
CodecType	[I/O] Sets/Gets the codec type.
ContentStatus	[O] Check current streaming status
Contrast	[I/O] Sets/Gets the contrast value of video configuration
ControlPort	[I/O] Sets/Gets the special control port
DeviceChipType	[O] Gets the device chip type
Duration	[O] Total playing time of current playback media
EnableASE	[I/O] Connect to IP device by streaming engine or not
EnableBorder	[I/O] Enable ActiveX control border
EndTime	[O] The time stamp in the playback media
Fps	[I/O] Sets/Gets the frame per second value of video configuration
FrameRateMode	[O] Returns current camera frame rate mode type
HttpPort	[I/O] Sets the special HTTP port
Hue	[I/O] Sets/Gets the hue value of video configuration
ID	[I/O] Sets/Gets the camera ID
錯誤! 找不到參照來源。	[O] Check if connected device is a quad device
LastRecFileSize	[O]Gets the last record file size
longLastRecFileSize	[O]Gets the last record file size in long
MediaChannel	[I/O] Sets/Gets the camera channel ID
MediaPassword	[I] Sets the password to access the media stream

MediaSource	[I/O] Identifies the source of the media stream
MediaType	[I/O] Sets/Gets the media type: Preview, Playback or PTZ status
MediaUsername	[I/O] Sets the user name of the media stream
MulticastIP	[I/O] Sets/Gets the multicast IP address
MulticastPort	[I/O] Sets/Gets the multicast port
Mute	[I] Turn on or off the PC sound device.
NetworkStatus	[O] Check if current network status
PreRecordTime	[I/O] Sets/Gets the pre-recording time before an event
PostRecordingTime	[I/O] Sets/Gets the post-recording time after an event
RecordType	[O] Returns current camera recording status
RegisterPort	[I/O] Gets/Sets the register port
ReplaceTimeCodeByLocalTime	[I] Replace time code by local time
Resolution	[I/O] Sets/Gets the resolution value of video configuration
RTSPPort	[I/O] Sets the special RTSP port
Saturation	[I/O] Sets/Gets the saturation value of video configuration
StreamingPort	[I/O] Sets/Gets the streaming port
StreamType	[I/O] Sets/Gets the streaming type
strLastRecFileSize	[O] Gets the last record file size in string
VariableF	[I/O] Sets/Gets the variable frame per second value of video configuration
Version	[O] Returns the SDK version
Volume	[I/O] Sets/Gets the PC sound Volume.
DeviceType	[I/O] Sets/Gets the device type value.
QuadDeviceMode	[I/O] Sets/Gets device mode value on quad device.

AutoReconnect

Description

[I/O] Sets/Gets whether certain streaming media reconnect automatically when connection lost

Syntax

AutoReconnect = *Input_value*

Return_value = **AutoReconnect** [*default: 0*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Input_value</i>	LONG	Sets the value of AutoStart flag						
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>AutoReconnect sets to no</td></tr><tr><td>1</td><td>AutoReconnect sets to yes</td></tr></table>	Value	Description	0	AutoReconnect sets to no	1	AutoReconnect sets to yes
Value	Description							
0	AutoReconnect sets to no							
1	AutoReconnect sets to yes							

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	LONG	Returns current settings value

Remarks

Default is no(0). Meaning that when network connection is broken, SDK will not re-connect the session automatically

Example

```
ObjStreaming.MediaType = 1;
ObjStreaming.ID = 1;
ObjStreaming.MediaSource = "192.168.0.100";
ObjStreaming.MediaUsername = "Admin";
ObjStreaming.MediaPassword = "123456";
ObjStreaming.AutoReconnect = 1;
ObjStreaming.Connect(0) ;
ObjStreaming.StartStream();
If (ObjStreaming.NetworkStatus != 0) {
    ObjStreaming.Play() ;
}
```

See Also

connect(), Play()

AutoReconnectInterval

Description

[I/O] Sets/Gets auto reconnect interval whether certain streaming media reconnect automatically when connection loss

Syntax

AutoReconnectInterval = *Input_Value*
Return_Value = **AutoReconnectInterval** [*default: 10*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Sets the value of interval

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns the value of interval

Remarks

Default is 10 second with reconnect interval.

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100";
objStreaming.MediaUsername = "Admin";
objStreaming.MediaPassword = "123456";
objStreaming.AutoReconnect = 1;
objStreaming.AutoReconnectInterval = 10; // 10 second will try to reconnect again
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

See Also

[Connect\(\)](#), [Play\(\)](#)

ASEControlPort

Description

[I/O] Sets/Gets the special control port for streaming engine

Syntax

```
ASEControlPort = Input_Value  
Return_Value = ASEControlPort
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Set the special control port for streaming engine

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current settings value for streaming engine

Remarks

NONE

Example

```
.....  
ObjStreaming.MediaSource = "192.168.0.100" ;  
ObjStreaming.MediaUsername = "Admin" ;  
ObjStreaming.MediaPassword = "123456" ;  
ObjStreaming.RegisterPort = 6000 ;  
ObjStreaming.ControlPort = 6001 ;  
ObjStreaming.StreamingPort = 6002 ;  
ObjStreaming.MulticastPort = 5000 ;  
ObjStreaming.HttpPort = 80 ;  
... ..  
ObjStreaming.ASEControlPort = 6001  
... ..  
objStreaming.Connect(0) ;  
objStreaming.StartStream();  
.....
```

See Also

RegisterPort, StreamingPort, MulticastPort, HttpPort

ASEMediaSource

Description

[I/O] Sets/Gets the source of the media stream for streaming engine.

Syntax

```
ASEMediaSource = "Input_value"  
"Return_value" = ASEMediaSource
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	BSTR	Sets the source of the media stream The source can be an IP address or hostname

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Returns current value stored in ASEMediaSource.

Remarks

NONE

Example

```
objStreaming.MediaType = 1;  
objStreaming.ID = 1;  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
  
objStreaming.ASEMediaSource = "172.16.1.178" ; // Streaming engine server address  
objStreaming.ASEMediaUserName = "Admin" ;  
objStreaming.ASEMediaPassword = "123456" ;  
objStreaming.EnableASE = TRUE ;  
  
objStreaming.Connect(0) ;  
objStreaming.StartStream();
```

```
If (ObjStreaming.NetworkStatus != 0) {  
    objStreaming.Play() ;  
}
```

See Also

[connect\(\)](#)

ASEMediaUserName

Description

[I/O] Sets the user name of the media stream for streaming engine

Syntax

ASEMediaUserName = *"Input_value"*
"Return_value" = **ASEMediaUserName**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	BSTR	Sets the user name of the media stream for streaming engine

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Gets the user name of the media stream for streaming engine

Remarks

NONE

Example

```
objStreaming.MediaType = 1;  
objStreaming.ID = 1;  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
  
objStreaming.ASEMediaSource = "172.16.1.178" ; // Streaming engine server address  
objStreaming.ASEMediaUserName = "Admin" ;  
objStreaming.ASEMediaPassword = "123456" ;  
objStreaming.EnableASE = TRUE ;  
  
objStreaming.Connect(0) ;  
objStreaming.StartStream();
```

```
If (ObjStreaming.NetworkStatus != 0) {  
    objStreaming.Play() ;  
}
```

See Also

[connect\(\)](#)

ASEMediaPassword

Description

[I/O] Sets/Gets the password of the media stream for streaming engine

Syntax

```
ASEMediaPassword = "Input_value"  
"Return_value" = ASEMediaPassword
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	BSTR	Sets the password of the media stream for streaming engine

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Gets the password of the media stream for streaming engine

Remarks

NONE

Example

```
objStreaming.MediaType = 1;  
objStreaming.ID = 1;  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
  
objStreaming.ASEMediaSource = "172.16.1.178" ; // Streaming engine server address  
objStreaming.ASEMediaUserName = "Admin" ;  
objStreaming.ASEMediaPassword = "123456" ;  
objStreaming.EnableASE = TRUE ;  
  
objStreaming.Connect(0) ;  
objStreaming.StartStream();  
  
If (objStreaming.NetworkStatus != 0) {
```

```
| objStreaming.Play() ;  
| }
```

See Also

[connect\(\)](#)

ASEStreamingPort

Description

[I/O] Sets/Gets the streaming port for streaming engine

Syntax

```
ASEStreamingPort = Input_Value  
Return_Value = ASEStreamingPort
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Set the special streaming port for streaming engine

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current settings value for streaming engine

Remarks

NONE

Example

```
.....  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
objStreaming.RegisterPort = 6000 ;  
objStreaming.ControlPort = 6001 ;  
objStreaming.StreamingPort = 6002 ;  
objStreaming.MulticastPort = 5000 ;  
objStreaming.HttpPort = 80 ;  
.....  
objStreaming.ASEStreamingPort = 6002 ;  
.....  
objStreaming.Connect(0) ;  
objStreaming.StartStream();  
.....
```

See Also

ControlPort, MulticastPort, HttpPort, MulticastPortRegisterPort, StreamingPort

BeginTime

Description

[0] Gets the time stamp in the playback media

Syntax

Return_Value = **BeginTime**

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns the time stamp in the playback media

Remarks

NONE

Example

```
objStreaming.MediaType = 2 ;
objStreaming.MediaSource = "C:\abc.raw" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

dwBeginTime = objStreaming.BeginTime ;
```

See Also

Play(), Duration,

BitRate

Description

[I/O] Sets/Gets the bit-rate value of video configuration

Syntax

BitRate = *Input_value*
Return_value = **BitRate**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>																																								
<i>Input_value</i>	LONG	Sets the bit-rate value of the video configuration																																								
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>28 Kbps</td></tr><tr><td>1</td><td>56 Kbps</td></tr><tr><td>2</td><td>128 Kbps</td></tr><tr><td>3</td><td>256 Kbps</td></tr><tr><td>4</td><td>384 Kbps</td></tr><tr><td>5</td><td>500 Kbps</td></tr><tr><td>6</td><td>750 Kbps</td></tr><tr><td>7</td><td>1 Mbps</td></tr><tr><td>8</td><td>1.2 Mbps</td></tr><tr><td>9</td><td>1.5 Mbps</td></tr><tr><td>10</td><td>2 Mbps</td></tr><tr><td>11</td><td>2.5 Mbps</td></tr><tr><td>12</td><td>3 Mbps</td></tr><tr><td>13</td><td>3.5 Mbps</td></tr><tr><td>14</td><td>4 Mbps</td></tr><tr><td>15</td><td>4.5 Mbps</td></tr><tr><td>16</td><td>5 Mbps</td></tr><tr><td>17</td><td>5.5 Mbps</td></tr><tr><td>18</td><td>6 Mbps</td></tr></table>	Value	Description	0	28 Kbps	1	56 Kbps	2	128 Kbps	3	256 Kbps	4	384 Kbps	5	500 Kbps	6	750 Kbps	7	1 Mbps	8	1.2 Mbps	9	1.5 Mbps	10	2 Mbps	11	2.5 Mbps	12	3 Mbps	13	3.5 Mbps	14	4 Mbps	15	4.5 Mbps	16	5 Mbps	17	5.5 Mbps	18	6 Mbps
Value	Description																																									
0	28 Kbps																																									
1	56 Kbps																																									
2	128 Kbps																																									
3	256 Kbps																																									
4	384 Kbps																																									
5	500 Kbps																																									
6	750 Kbps																																									
7	1 Mbps																																									
8	1.2 Mbps																																									
9	1.5 Mbps																																									
10	2 Mbps																																									
11	2.5 Mbps																																									
12	3 Mbps																																									
13	3.5 Mbps																																									
14	4 Mbps																																									
15	4.5 Mbps																																									
16	5 Mbps																																									
17	5.5 Mbps																																									
18	6 Mbps																																									

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns the bit-rate value of the video configuration

Remarks

From 3.5M to 6M are available in PaltformT encoders only.

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

dwBitRate = objStreaming.BitRate    ' get current bit rate setting
```

See Also

Resolution, Fps

Brightness

Description

[I/O] Sets/Gets the brightness value of video configuration

Syntax

Brightness = *Input_Value*
Return_Value = **Brightness**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>				
<i>Input_value</i>	LONG	Sets the brightness value of the video configuration				
		<table><tr><th>Minimum</th><th>Maximum</th></tr><tr><td>0</td><td>100</td></tr></table>	Minimum	Maximum	0	100
Minimum	Maximum					
0	100					

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current brightness value in video configuration

Remarks

NONE

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

```
objStreaming.Brightness = 30 ;  
dwBrightness = objStreaming.Brightness ;
```

See Also

Hue, Contrast

CodecType

Description

[I/O] Sets/Gets the codec type.

Syntax

`CodecType = Input_value`
`Return_value = CodecType`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>														
<i>Input_value</i>	LONG	Sets the value of codec type. <table><tr><th>value</th><th>description</th></tr><tr><td>0</td><td>XVID</td></tr><tr><td>1</td><td>FFMPEG</td></tr><tr><td>2</td><td>PCI51</td></tr><tr><td>3</td><td>IPP</td></tr><tr><td>4</td><td>MJPEG</td></tr><tr><td>5</td><td>H.264</td></tr></table>	value	description	0	XVID	1	FFMPEG	2	PCI51	3	IPP	4	MJPEG	5	H.264
value	description															
0	XVID															
1	FFMPEG															
2	PCI51															
3	IPP															
4	MJPEG															
5	H.264															

Returns

<i>NO</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Return current setting value.

Remarks

NONE

Example

```
ObjStreaming.MediaType = 1;
ObjStreaming.ID = 1;
ObjStreaming.MediaSource = "192.168.0.100" ;
ObjStreaming.MediaUsername = "Admin" ;
ObjStreaming.MediaPassword = "123456" ;
ObjStreaming.AutoReconnect = 1 ;
ObjStreaming.Connect(0) ;
ObjStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
```

```
Alert(ObjStreaming.CodecType);  
}
```

See Also

mediaType

ContentStatus

Description

[0] Gets current streaming status

Syntax

Return_value = **ContentStatus** [*default: 0*]

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>										
<i>Return_value</i>	LONG	Check current streaming status										
		<table><tr><th>Value</th><th>Define</th></tr><tr><td>0</td><td>_STREAMING_MODE_STOP</td></tr><tr><td>1</td><td>_STREAMING_MODE_START</td></tr><tr><td>2</td><td>_STREAMING_MODE_PLAY</td></tr><tr><td>3</td><td>_STREAMING_MODE_PAUSE</td></tr></table>	Value	Define	0	_STREAMING_MODE_STOP	1	_STREAMING_MODE_START	2	_STREAMING_MODE_PLAY	3	_STREAMING_MODE_PAUSE
Value	Define											
0	_STREAMING_MODE_STOP											
1	_STREAMING_MODE_START											
2	_STREAMING_MODE_PLAY											
3	_STREAMING_MODE_PAUSE											

Remarks

NONE

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.Connect(0) ;

If (objStreaming.NetworkStatus != 0)
{
    ContentStatus = objStreaming.ContentStatus;
    if(ContentStatus != _STREAMING_MODE_START)
        objStreaming.StartStream();

    objStreaming.Play() ;
}
```

See Also

Contrast

Description

[I/O] Sets/Gets the contrast value of video configuration

Syntax

Contrast = *Input_Value*
Return_Value = **Contrast**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>				
<i>Input_value</i>	LONG	Sets the contrast value of the video configuration				
		<table><tr><th>Minimum</th><th>Maximum</th></tr><tr><td>0</td><td>100</td></tr></table>	Minimum	Maximum	0	100
Minimum	Maximum					
0	100					

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current contrast value in video configuration

Remarks

NONE

Example

```
objStreaming.MediaType = 1;  
objStreaming.ID = 1;  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
objStreaming.AutoReconnect = 1 ;  
objStreaming.Connect(0) ;  
objStreaming.StartStream();  
  
If (objStreaming.NetworkStatus != 0) {  
    objStreaming.Play() ;  
}
```

```
objStreaming.Contrast = 30 ;  
dwContrast = objStreaming.Contrast ;
```

See Also

Brightness, Hue

ControlPort

Description

[I/O] Sets/Gets the special control port

Syntax

```
ControlPort = Input_Value  
Return_Value = ControlPort
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Set the special control port

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current settings value

Remarks

NONE

Example

```
.....  
ObjStreaming.MediaSource = "192.168.0.100" ;  
ObjStreaming.MediaUsername = "Admin" ;  
ObjStreaming.MediaPassword = "123456" ;  
ObjStreaming.RegisterPort = 6000 ;  
ObjStreaming.ControlPort = 6001 ;  
ObjStreaming.StreamingPort = 6002 ;  
ObjStreaming.MulticastPort = 5000 ;  
ObjStreaming.HttpPort = 80 ;  
  
objStreaming.Connect(0) ;  
objStreaming.StartStream();  
  
.....
```

See Also

RegisterPort, StreamingPort, MulticastPort, HttpPort

DeviceChipType

Description

[0] Gets the device chip type

Syntax

Return_value = DeviceChipType [default: 0]

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns the value of device chip type.

Remarks

NONE

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.DeviceChipType = 1
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

See Also

Duration

Description

[o] Gets total playing second time of current playback media

Syntax

Return_Value = **Duration**

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns the length of the playing streaming media

Remarks

NONE

Example

```
objStreaming.MediaType = 2 ;
objStreaming.MediaSource = "C:\abc.raw" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

lTotalLength = objStreaming.Duration ;
```

See Also

BeginTime, **Play()**

EnableASE

Description

[I/O] Connect to IP device by streaming engine or not

Syntax

```
EnableASE = Input_Value  
Return_Value = EnableASE [ default: FALSE ]
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	Bool	Connect to IP device by streaming engine.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	Bool	This connection is using streaming engine

Remarks

NONE

Example

```
objStreaming.ASEMediaSource = "172.16.1.178" ; // Streaming engine server address  
objStreaming.ASEMediaUserName = "Admin" ;  
objStreaming.ASEMediaPassword = "123456" ;  
objStreaming.EnableASE = TRUE ;
```

See Also

EnableBorder

Description

[I/O] Sets/Gets the ActiveX control border status.

Syntax

```
EnableBorder = Input_Value  
Return_Value = EnableBorder [ default: FALSE ]
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	BOOL	Set ActiveX control border enabled.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BOOL	Get ActiveX control border enabled status

Remarks

NONE

Example

```
objStreaming.EnableBorder = TRUE ;
```

See Also

EndTime

Description

[o] Gets the time stamp in the playback media

Syntax

Return_Value = EndTime

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	Long	Returns the time stamp in the playback media

Remarks

NONE

Example

```
objStreaming.MediaType = 2 ;  
objStreaming.MediaSource = "C:\abc.raw" ;  
objStreaming.Connect(0) ;  
objStreaming.StartStream();  
  
If (objStreaming.NetworkStatus != 0) {  
    objStreaming.Play() ;  
}  
  
dwBeginTime = objStreaming.EndTime ;
```

See Also

Play(), Duration

Fps

Description

[I/O] Sets/Gets the frame per second value of video configuration

Syntax

Fps = *Input_value*
Return_value = **Fps**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	LONG	Sets the frame per second value of the video configuration Under NTSC mode, the available frame rate are: 30, 6, 3, 1 Under PAL mode, the available frame rate are: 25, 5, 3, 1



IMPORTANT: To set this property, it only works under Variable Frame Rate settings. (Please check Web Configuration setting on this).

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns current frame per second value in video configuration

Remarks

If Fps is set to 0, then the video will not be displayed.

Example

```
objStreaming.MediaType = 1;  
objStreaming.ID = 1;  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
objStreaming.Connect(0) ;  
objStreaming.StartStream();
```

```
If (ObjStreaming.NetworkStatus != 0) {  
    objStreaming.Play() ;  
}  
  
if (objStreaming.FpsMode == 1) {  
    objStreaming.Fps = 30 ;  
}
```

See Also

BitRate, Resolution

FrameRateMode

Description

[0] Gets current camera frame rate mode type

Syntax

Return_Value = **FrameRateMode**

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns current camera FPS mode type s						
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>1</td><td>Is constant frame rate mode</td></tr><tr><td>2</td><td>Is variable frame rate mode</td></tr></table>	Value	Description	1	Is constant frame rate mode	2	Is variable frame rate mode
Value	Description							
1	Is constant frame rate mode							
2	Is variable frame rate mode							

Remarks

Please check Web Configurator (firmware) User's Manual for Variable and Constant Frame Rate settings.

Example

```
.....
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

if(objStreaming.FrameRateMode == 0 )
    MessageBox("Constant FPS Mode");
else
    MessageBox("Variable FPS Mode");
```

See Also

HttpPort

Description

[I/O] Sets/Gets the special Http port

Syntax

HttpPort = *Input_value*
Return_value = **HttpPort**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	Long	Set the special Http port

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	Long	Returns current settings value

Remarks

NONE

Example

```
.....  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
objStreaming.RegisterPort = 6000 ;  
objStreaming.ControlPort = 6001 ;  
objStreaming.StreamingPort = 6002 ;  
objStreaming.MulticastPort = 5000 ;  
objStreaming.HttpPort = 80 ;  
  
objStreaming.Connect(0) ;  
objStreaming.StartStream();  
.....
```

See Also

RegisterPort, StreamingPort, MulticastPort, ControlPort

Hue

Description

[I/O] Sets/Gets the hue value of video configuration

Syntax

Hue = *Input_value*
Return_value = **Hue**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>				
<i>Input_value</i>	LONG	Sets the hue value of the video configuration				
		<table><tr><th>Minimum</th><th>Maximum</th></tr><tr><td>0</td><td>100</td></tr></table>	Minimum	Maximum	0	100
Minimum	Maximum					
0	100					

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns current hue value in video configuration

Remarks

NONE

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

objStreaming.Hue = 30 ;
dwHue = objStreaming.Hue ;
```

See Also

Contrast, Brightness

ID

Description

[I/O] Sets/Gets the camera ID

Syntax

ID = *Input_value*
Return_value = *ID*

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>				
<i>Input_value</i>	LONG	Sets the camera ID				
		<table><tr><th>Minimum</th><th>Maximum</th></tr><tr><td>1</td><td>255</td></tr></table>	Minimum	Maximum	1	255
Minimum	Maximum					
1	255					

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Gets current camera ID

Remarks

This is used in event as callback ID. For example, motion detection callback needs to assign an ID to identify which video server submit this event.

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;      ' ID 1 to identify number 1 of the object
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

See Also

LastRecFileSize

Description

[I/O] Sets/Gets the last record file size

Syntax

```
LastRecFileSize = Input_value  
Return_value = LastRecFileSize
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	ULONGLONG	Sets the last record file size

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	ULONGLONG	Gets the last record file size

Remarks

NONE

Example

```
objStreaming.LastRecFileSize;
```

See Also

mediaUsername, Connect()

longLastRecFileSize

Description

[I/O] Sets/Gets the last record file size in long

Syntax

```
longLastRecFileSize = Input_value  
Return_value = longLastRecFileSize
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	LONG	Sets the last record file size

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Gets the last record file size

Remarks

NONE

Example

```
objStreaming.longLastRecFileSize;
```

See Also

MediaUsername, Connect()

MediaChannel

Description

[I/O] Sets/Gets the camera channel ID

Syntax

```
MediaChannel = Input_Value  
Return_Value = MediaChannel
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Sets the camera channel ID

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Gets the camera channel ID

Remarks

NONE

Example

```
ObjStreaming.MediaType = 1;  
ObjStreaming.ID = 1;  
ObjStreaming.MediaSource = "192.168.0.100" ;  
ObjStreaming.MediaChannel = 1 ;  
ObjStreaming.MediaUsername = "Admin" ;  
ObjStreaming.MediaPassword = "123456" ;  
ObjStreaming.Connect(0) ;  
ObjStreaming.StartStream();  
  
If (ObjStreaming.NetworkStatus != 0) {  
    ObjStreaming.Play() ;  
}
```

See Also

MediaUsername, Connect()

MediaPassword

Description

[1] Sets the password to access the media stream

Syntax

MediaPassword = "*Input_Value*"

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	BSTR	Sets the password to access the media source

Returns

NONE

Remarks

NONE

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

See Also

MediaUsername, Connect()

MediaSource

Description

[I/O] Sets/Gets the source of the media stream

Syntax

MediaSource = "*Input_value*"

"Return_value" = **MediaSource**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	BSTR	Sets the source of the media stream The source can be an IP address or a filename

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Returns current value stored in MediaSource.

Remarks

NONE

Example

```
ObjStreaming.MediaType = 1;
ObjStreaming.ID = 1;
ObjStreaming.MediaSource = "192.168.0.100" ;
ObjStreaming.MediaUsername = "Admin" ;
ObjStreaming.MediaPassword = "123456" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

See Also

[connect\(\)](#)

MediaType

Description

[I/O] Sets/Gets the media type: Preview, Playback or PTZ status

Syntax

```
MediaType = Input_value  
Return_value = MediaType [ default: 0 ]
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>																				
<i>Input_value</i>	LONG	Sets the value of MediaType flag <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Preview but no control</td></tr><tr><td>1</td><td>Preview with Unicast</td></tr><tr><td>2</td><td>Playback</td></tr><tr><td>3</td><td>Send PTZ command, but no streaming</td></tr><tr><td>4</td><td>Preview with Multicast</td></tr><tr><td>5</td><td>Preview with Multicast, but no control</td></tr><tr><td>6</td><td>Multiplayback</td></tr><tr><td>7</td><td>RTSP</td></tr><tr><td>8</td><td>AHTTP</td></tr></table>	Value	Description	0	Preview but no control	1	Preview with Unicast	2	Playback	3	Send PTZ command, but no streaming	4	Preview with Multicast	5	Preview with Multicast, but no control	6	Multiplayback	7	RTSP	8	AHTTP
Value	Description																					
0	Preview but no control																					
1	Preview with Unicast																					
2	Playback																					
3	Send PTZ command, but no streaming																					
4	Preview with Multicast																					
5	Preview with Multicast, but no control																					
6	Multiplayback																					
7	RTSP																					
8	AHTTP																					

Returns

Name	Type	Description
Return_value	LONG	Returns current setting value.

Remarks

NONE

Example

```
ObjStreaming.MediaType = 1;  
ObjStreaming.ID = 1;  
ObjStreaming.MediaSource = "192.168.0.100" ;  
ObjStreaming.MediaUsername = "Admin" ;  
ObjStreaming.MediaPassword = "123456" ;  
ObjStreaming.AutoReconnect = 1 ;  
ObjStreaming.Connect(0) ;  
ObjStreaming.StartStream();  
  
If (ObjStreaming.NetworkStatus != 0) {
```

```
    objStreaming.Play() ;  
}
```

See Also

Play()

MediaUsername

Description

[I/O] Sets/Gets the user name of the media stream

Syntax

MediaUsername = *"Input_Value"*
"Return_Value" = **MediaUsername**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	BSTR	Sets the user name of the media stream

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Gets the user name of the media stream

Remarks

NONE

Example

```
objStreaming.MediaType = 1;  
objStreaming.ID = 1;  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
objStreaming.Connect(0) ;  
objStreaming.StartStream();  
  
If (objStreaming.NetworkStatus != 0) {  
    objStreaming.Play() ;  
}
```

See Also

MediaPassword, Connect()

MulticastIP

Description

[I/O] Sets/Gets the multicast IP address

Syntax

MulticastIP = "Input_value"

"Return_value" = **MulticastIP**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	BSTR	Sets the multicast IP address of the media stream

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Gets the multicast IP address of the media stream

Remarks

If you set multicast IP, the streaming preview mode will be using multicast mode otherwise will be use unicast mode.

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MulticastIP = "228.5.6.1" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

See Also

Connect(), **MulticastPort**

Mute

Description

[I/O] Sets/Gets the audio to turn on or off.

Syntax

Mute = *Input_value*

Return_value = **Mute** [*default: 0*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Input_value</i>	LONG	Specifies the flag to decode frames						
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Audio Out</td></tr><tr><td>1</td><td>Audio Mute</td></tr></table>	Value	Description	0	Audio Out	1	Audio Mute
Value	Description							
0	Audio Out							
1	Audio Mute							

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns the current setting value.

Remarks

NONE

Example

```
.....
```

```
objStreaming.Play() ;  
objStreaming.Mute = 1 ;  
objStreaming.SetVolume(30) ;
```

See Also

MulticastPort

Description

[I/O] Sets/Gets the multicast port

Syntax

```
MulticastPort = Input_Value  
Return_Value = MulticastPort
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Set the special Multicast port

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current setting value

Remarks

NONE

Example

```
.....  
ObjStreaming.MediaSource = "192.168.0.100" ;  
ObjStreaming.MediaUsername = "Admin" ;  
ObjStreaming.MediaPassword = "123456" ;  
ObjStreaming.RegisterPort = 6000 ;  
ObjStreaming.ControlPort = 6001 ;  
ObjStreaming.StreamingPort = 6002 ;  
ObjStreaming.MulticastPort = 5000 ;  
ObjStreaming.HttpPort = 80 ;  
  
objStreaming.Connect(0) ;  
objStreaming.StartStream();
```

See Also

connect(), MulticastIP

NetworkStatus

Description

[0] Gets current network status

Syntax

Return_value = **NetworkStatus** [*default: 0*]

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>												
<i>Return_value</i>	LONG	Check if current status did connected or not												
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Disconnected</td></tr><tr><td>1</td><td>Conenct control Only</td></tr><tr><td>2</td><td>Conenct Streaming</td></tr><tr><td>3</td><td>Conenct control and Streaming</td></tr><tr><td>4</td><td>Disconnecting</td></tr></table>	Value	Description	0	Disconnected	1	Conenct control Only	2	Conenct Streaming	3	Conenct control and Streaming	4	Disconnecting
Value	Description													
0	Disconnected													
1	Conenct control Only													
2	Conenct Streaming													
3	Conenct control and Streaming													
4	Disconnecting													

Remarks

NONE

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

See Also

PreRecordTime

Description

[I/O] Sets/Gets the pre-recording time before an event

Syntax

PreRecordTime = *Input_value*
Return_value = **PreRecordTime**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>				
<i>Input_value</i>	LONG	Sets the pre-recording time before an event				
		<table><tr><th>Minimum</th><th>Maximum</th></tr><tr><td>5 (Sec)</td><td>30 (Sec)</td></tr></table>	Minimum	Maximum	5 (Sec)	30 (Sec)
Minimum	Maximum					
5 (Sec)	30 (Sec)					

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Gets the value of the pre-recording time before an event

Remarks

NONE

Example

```
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
if (objStreaming.ContentStatus == 1) {
    objStreaming.PreRecordTime = 10 ;
    objStreaming.PostRecordingTime = 10 ;
    objStreaming.StartAlarmRecord("C:\ABC.raw");
}
```

See Also

PostRecordingTime

PostRecordingTime

Description

[I/O] Sets/Gets the post-recording time after an event

Syntax

PostRecordingTime = *Input_value*
Return_value = **PostRecordingTime**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>				
<i>Input_value</i>	LONG	Sets the post-recording time after an startalarmrecord				
		<table><tr><th>Minimum</th><th>Maximum</th></tr><tr><td>5 (Sec)</td><td>30 (Sec)</td></tr></table>	Minimum	Maximum	5 (Sec)	30 (Sec)
Minimum	Maximum					
5 (Sec)	30 (Sec)					

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns the value of the post-recording time after an startalarmrecord

Remarks

NONE

Example

```
.....  
objStreaming.Connect(0) ;  
objStreaming.StartStream();  
  
If (ObjStreaming.NetworkStatus != 0) {  
    objStreaming.Play() ;  
}  
  
if (objStreaming.ContentStatus == 1) {  
    objStreaming.PreRecordTime = 10 ;  
    objStreaming.PostRecordingTime = 10 ;  
    objStreaming.StartAlarmRecord("C:\ABC.raw");  
}
```

See Also

PreRecordTime

RecordType

Description

[I/O] Sets/Gets the current camera recording status

Syntax

RecordType = *Input_value*

Return_value = **RecordType** [*default: 0*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>								
<i>Input_value</i>	LONG	Sets current camera recording status								
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Record Raw file</td></tr><tr><td>1</td><td>Record AVI file</td></tr><tr><td>2</td><td>Record RAW + IDX file.</td></tr></table>	Value	Description	0	Record Raw file	1	Record AVI file	2	Record RAW + IDX file.
Value	Description									
0	Record Raw file									
1	Record AVI file									
2	Record RAW + IDX file.									

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns current camera recording status

Remarks

NONE

Example

```
.....
objStreaming.Connect(0);
objStreaming.StartStream();
objStreaming.RecordType = 0;
objStreaming.Play() ;

if (objStreaming.ContentStatus == 1) {
    objStreaming.StartRecord("C:\\ABC.raw");
}
```

See Also

RegisterPort

Description

[I/O] Gets/Sets the register port

Syntax

RegisterPort = *Input_Value*
Return_Value = **RegisterPort**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Set the special register port

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current settings value

Remarks

NONE

Example

```
.....  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
objStreaming.RegisterPort = 6000 ;  
objStreaming.ControlPort = 6001 ;  
objStreaming.StreamingPort = 6002 ;  
objStreaming.MulticastPort = 5000 ;  
objStreaming.HttpPort = 80 ;  
objStreaming.Connect(0) ;  
objStreaming.StartStream();
```

See Also

ControlPort, HttpPort, MulticastPort, RegisterPort, StreamingPort

ReplaceTimeCodeByLocalTime

Description

[I/O] Sets/Gets the replace time code by local time

Syntax

`ReplaceTimeCodeByLocalTime = Input_Value`

`Return_Value = ReplaceTimeCodeByLocalTime [default: 1]`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Input_value</i>	LONG	Sets replace time code by local time						
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Disable</td></tr><tr><td>1</td><td>Enable</td></tr></table>	Value	Description	0	Disable	1	Enable
Value	Description							
0	Disable							
1	Enable							

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns the replace time code by local time

Remarks

NONE

Example

```
.....
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.RegisterPort = 6000 ;
objStreaming.ControlPort = 6001 ;
objStreaming.StreamingPort = 6002 ;
objStreaming.MulticastPort = 5000 ;
objStreaming.HttpPort = 80 ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

objStreaming.ReplaceTimeCodeByLocalTime = 1
```

See Also

Resolution

Description

[0] Gets the resolution value of video configuration

Syntax

Return_Value = **Resolution** [*default: 0*]

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>																																												
<i>Return_value</i>	Long	Returns current resolution value in video configuration																																												
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>NTSC 720x480</td></tr><tr><td>1</td><td>NTSC 352x240</td></tr><tr><td>2</td><td>NTSC 160x112</td></tr><tr><td>3</td><td>PAL 720x576</td></tr><tr><td>4</td><td>PAL 352x288</td></tr><tr><td>5</td><td>PAL 176x144</td></tr><tr><td>6</td><td>NTSC 176x120</td></tr><tr><td>64</td><td>NTSC 640x480</td></tr><tr><td>65</td><td>NTSC 1280x720</td></tr><tr><td>66</td><td>NTSC 1280x960</td></tr><tr><td>67</td><td>NTSC 1280x1024</td></tr><tr><td>68</td><td>NTSC 1600x1200</td></tr><tr><td>69</td><td>NTSC 1920x1080</td></tr><tr><td>70</td><td>NTSC 320x240</td></tr><tr><td>71</td><td>NTSC 160x120</td></tr><tr><td>72</td><td>NTSC 2032x1920</td></tr><tr><td>73</td><td>NTSC 1280x352</td></tr><tr><td>74</td><td>NTSC 1920x1072</td></tr><tr><td>192</td><td>PAL 640x480</td></tr><tr><td>198</td><td>PAL 320x240</td></tr><tr><td>199</td><td>PAL 160x112</td></tr></table>	Value	Description	0	NTSC 720x480	1	NTSC 352x240	2	NTSC 160x112	3	PAL 720x576	4	PAL 352x288	5	PAL 176x144	6	NTSC 176x120	64	NTSC 640x480	65	NTSC 1280x720	66	NTSC 1280x960	67	NTSC 1280x1024	68	NTSC 1600x1200	69	NTSC 1920x1080	70	NTSC 320x240	71	NTSC 160x120	72	NTSC 2032x1920	73	NTSC 1280x352	74	NTSC 1920x1072	192	PAL 640x480	198	PAL 320x240	199	PAL 160x112
Value	Description																																													
0	NTSC 720x480																																													
1	NTSC 352x240																																													
2	NTSC 160x112																																													
3	PAL 720x576																																													
4	PAL 352x288																																													
5	PAL 176x144																																													
6	NTSC 176x120																																													
64	NTSC 640x480																																													
65	NTSC 1280x720																																													
66	NTSC 1280x960																																													
67	NTSC 1280x1024																																													
68	NTSC 1600x1200																																													
69	NTSC 1920x1080																																													
70	NTSC 320x240																																													
71	NTSC 160x120																																													
72	NTSC 2032x1920																																													
73	NTSC 1280x352																																													
74	NTSC 1920x1072																																													
192	PAL 640x480																																													
198	PAL 320x240																																													
199	PAL 160x112																																													

Remarks

NONE

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

dwResolution = objStreaming.Resolution
```

See Also

Fps, BitRate

RTSPPort

Description

[I/O] Sets/Gets the special RTSP port

Syntax

```
RTSPPort = Input_value  
Return_value = RTSPPort
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	LONG	Set the special RTSP port

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns current settings value

Remarks

NONE

Example

```
.....  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
objStreaming.RegisterPort = 6000 ;  
objStreaming.ControlPort = 6001 ;  
objStreaming.StreamingPort = 6002 ;  
objStreaming.MulticastPort = 5000 ;  
objStreaming.HttpPort = 80 ;  
objStreaming.RTSPPort = 554 ;  
  
objStreaming.Connect(0) ;  
objStreaming.StartStream();
```

See Also

RegisterPort, StreamingPort, MulticastPort, ControlPort

Saturation

Description

[I/O] Sets/Gets the saturation value of video configuration

Syntax

saturation = *Input_Value*

Return_Value = **Saturation** [*default: 50*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>				
<i>Input_Value</i>	Long	Sets the saturation value of the video configuration				
		<table><tr><th>Minimum</th><th>Maximum</th></tr><tr><td>0</td><td>100</td></tr></table>	Minimum	Maximum	0	100
Minimum	Maximum					
0	100					

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	Long	Returns current saturation value in video configuration

Remarks

NONE

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

```
objStreaming.Saturation = 50  
dwsaturation = objStreaming.Saturation
```

See Also

Hue , Brightness

StreamingPort

Description

[I/O] Sets/Gets the streaming port

Syntax

```
StreamingPort = Input_Value  
Return_Value = StreamingPort
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Set the special streaming port

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current settings value

Remarks

NONE

Example

```
.....  
ObjStreaming.MediaSource = "192.168.0.100" ;  
ObjStreaming.MediaUsername = "Admin" ;  
ObjStreaming.MediaPassword = "123456" ;  
ObjStreaming.RegisterPort = 6000 ;  
ObjStreaming.ControlPort = 6001 ;  
ObjStreaming.StreamingPort = 6002 ;  
ObjStreaming.MulticastPort = 5000 ;  
ObjStreaming.HttpPort = 80 ;  
  
ObjStreaming.Connect(0) ;  
ObjStreaming.StartStream();  
.....
```

See Also

ControlPort, MulticastPort, HttpPort, MulticastPortRegisterPort, StreamingPort

StreamType

Description

[I/O] Sets/Gets the streaming type

Syntax

```
StreamType = Input_Value  
Return_Value = StreamType
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Set the special streaming type

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current settings value

Remarks

NONE

Example

```
.....  
ObjStreaming.MediaSource = "192.168.0.100" ;  
ObjStreaming.MediaUsername = "Admin" ;  
ObjStreaming.MediaPassword = "123456" ;  
ObjStreaming.RegisterPort = 6000 ;  
ObjStreaming.ControlPort = 6001 ;  
ObjStreaming.StreamingPort = 6002 ;  
ObjStreaming.StreamingType = 1 ;  
ObjStreaming.MulticastPort = 5000 ;  
ObjStreaming.HttpPort = 80 ;  
  
objStreaming.Connect(0) ;  
objStreaming.StartStream();
```

See Also

ControlPort, MulticastPort, HttpPort, MulticastPortRegisterPort, StreamingPort

strLastRecFileSize

Description

[o] Gets the last record file size in string

Syntax

"Return_Value" = **strLastRecFileSize**

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Returns the last record file size

Remarks

NONE

Example

```
objStreaming.strLastRecFileSize;
```

See Also

MediaUsername, Connect()

VariableFPS

Description

[I/O] Sets/Gets the variable frame per second value of video configuration

Syntax

variableFPS = *Input_value*
Return_value = **variableFPS**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	LONG	Sets the variable frame per second value of the video configuration

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns current variable frame per second value in video configuration

Remarks

If Fps is set to 0, then the video will not be displayed.

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
if (objStreaming.FpsMode == 1) {
    objStreaming.VariableFPS = 30 ;
}
```

See Also

BitRate, Resolution

Version

Description

[0] Gets the SDK version

Syntax

Return_Value = **Version**

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Returns the SDK version

Remarks

NONE

Example

```
// sample by javascript
alert("The current version is :" + objStreaming.Version );
```

See Also

Volume

Description

[I/O] Sets/Gets the audio volume of the PC sound device.

Syntax

```
volume = Input_value  
Return_value = volume
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>				
<i>Input_value</i>	LONG	Sets the audio volume of the PC sound device. <table><tr><th>Minimum</th><th>Maximum</th></tr><tr><td>0</td><td>100</td></tr></table>	Minimum	Maximum	0	100
Minimum	Maximum					
0	100					

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Gets the audio volume of the PC sound device.

Remarks

NONE

Example

```
.....  
  
objStreaming.Play() ;  
objStreaming.Mute = 1 ;  
objStreaming.volume = 30 ;
```

See Also

Mute

DeviceType

Description

[I/O] Sets/Gets the device type value.

Syntax

DeviceType = *Input_Value*

Return_Value = **DeviceType** [*default*: _SINGLE_CHANNEL_VIDEO_SERVER]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>										
<i>Input_value</i>	DeviceType	Sets the device type. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>_SINGLE_CHANNEL_VIDEO_SERVER</td></tr><tr><td>1</td><td>_ACD2000Q_VIDEO_SERVER</td></tr><tr><td>2</td><td>_SED2300Q_VIDEO_SERVER</td></tr><tr><td>3</td><td>_AUTO_DETECT</td></tr></table>	Value	Description	0	_SINGLE_CHANNEL_VIDEO_SERVER	1	_ACD2000Q_VIDEO_SERVER	2	_SED2300Q_VIDEO_SERVER	3	_AUTO_DETECT
Value	Description											
0	_SINGLE_CHANNEL_VIDEO_SERVER											
1	_ACD2000Q_VIDEO_SERVER											
2	_SED2300Q_VIDEO_SERVER											
3	_AUTO_DETECT											

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	DeviceType	Returns current device type.

Remarks

NONE

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.DeviceType = _ACD2000Q_VIDEO_SERVER;
objStreaming.Connect(0) ;
objStreaming.StartStream();
```

See Also

QuadDeviceMode

Description

[I/O] Sets/Gets the device mode value on quad device.

Syntax

QuadDeviceMode = *Input_value*

Return_value = **QuadDeviceMode** [*default: SINGLE_MODE*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>										
<i>Input_value</i>	QuadDeviceMode	Sets the device mode value on quad device. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>QUAD_MODE</td></tr><tr><td>1</td><td>SINGLE_MODE</td></tr><tr><td>2</td><td>SEQUENTIAL_MODE</td></tr><tr><td>3</td><td>AUTO_DETECT</td></tr></table>	Value	Description	0	QUAD_MODE	1	SINGLE_MODE	2	SEQUENTIAL_MODE	3	AUTO_DETECT
Value	Description											
0	QUAD_MODE											
1	SINGLE_MODE											
2	SEQUENTIAL_MODE											
3	AUTO_DETECT											

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	QuadDeviceMode	Returns current device mode.

Remarks

NONE

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.QuadDeviceMode = QUAD_MODE;
objStreaming.Connect(0) ;
objStreaming.StartStream();
```

See Also

TCPVideoStreamID

Description

[I/O] Sets/Gets the device TCPVideoStreamID value. (Only for TCP 2.0)

Syntax

TCPVideoStreamID = *Input_Value*

Return_Value = **TCPVideoStreamID** [*default: 0*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	ULONG	Sets TCPVideoStreamID value. 0 ~ 255 for 1~ 256 video track

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	ULONG	Returns current TCPVideoStreamID.

Remarks

NONE

Example

```
ObjStreaming.MediaType = 1;  
ObjStreaming.ID = 1;  
ObjStreaming.MediaSource = "192.168.0.100" ;  
ObjStreaming.MediaUsername = "Admin" ;  
ObjStreaming.MediaPassword = "123456" ;  
ObjStreaming.MediaChannel = 1;  
ObjStreaming.TCPVideoStreamID = 0;  
ObjStreaming.Connect(0) ;  
ObjStreaming.StartStream();
```

See Also

RTPVideoTrackNumber

Description

[I/O] Sets/Gets the RTPVideoTrackNumber value. (Only for RTP)

Syntax

```
RTPVideoTrackNumber = Input_value  
Return_value = RTPVideoTrackNumber [ default: 0 ]
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	ULONG	Sets the RTPVideo track number value. 0 : Will be use 1 st video track. 1~255 : Specify video track.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	ULONG	Returns current setting value.

Remarks

NONE

Example

```
objStreaming.MediaType = 7;    //RTP  
objStreaming.ID = 1;  
objStreaming.MediaSource = "192.168.0.100" ;  
objStreaming.MediaUsername = "Admin" ;  
objStreaming.MediaPassword = "123456" ;  
objStreaming.MediaChannel = 1;  
objStreaming.RTPVideoTrackNumber = 0;  
objStreaming.RTPAudioTrackNumber = 0;  
objStreaming.Connect(0) ;  
objStreaming.StartStream();
```

See Also

RTPAudioTrackNumber

Description

[I/O] Sets/Gets the RTPAudioTrackNumber value. (Only for RTP)

Syntax

RTPAudioTrackNumber = *Input_value*
Return_value = **RTPAudioTrackNumber** [*default: 0*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	ULONG	Sets the RTPAudio track number value. 0 : Will be use 1 st audio track. 1~255 : Specify audio track.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	ULONG	Returns current setting value.

Remarks

NONE

Example

```
objStreaming.MediaType = 7;    //RTP
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.MediaChannel = 1;
objStreaming.RTPVideoTrackNumber = 0;
objStreaming.RTPAudioTrackNumber = 0;
objStreaming.Connect(0) ;
objStreaming.StartStream();
```

See Also

ConnectTimeout

Description

[I/O] Sets/Gets the connect timeout value.

Syntax

ConnectTimeout = *Input_Value*

Return_Value = **ConnectTimeout** [*default: 5*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Sets the connect timeout value.(seconds)

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current value.(seconds)

Remarks

NONE

UI Property

<i>Name</i>	<i>Description</i>
Caption	[I/O] Sets/Gets the caption of the window
StretchToFit	[I/O] stretch the video window to display window

Caption

Description

[I/O] Sets/Gets the caption of the window

Syntax

`Caption = "Input_value"`
`"Return_value" = Caption`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	BSTR	Sets the caption of the media window

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Returns current caption of the media window

Remarks

Notes: In preview mode, the caption always control and display by timer, about detail you can refer Desc property

Example

```
objStreaming.MediaType = 1;
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.Caption = "Lobby" ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

See Also

Play()

StretchToFit

Description

[I/O] Sets/Gets the display window activate or de-activate

Syntax

```
StretchToFit = Input_Value  
Return_Value = StretchToFit
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Input_value</i>	LONG	Activate or de-activate current media window						
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Do not Stretch to display window</td></tr><tr><td>1</td><td>Stretch to display window</td></tr></table>	Value	Description	0	Do not Stretch to display window	1	Stretch to display window
Value	Description							
0	Do not Stretch to display window							
1	Stretch to display window							

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current media window is activate or de-activate.

Remarks

NONE

Example

```
objStreaming.StretchToFit = 1 ;
```

See Also

PTZ Property

<i>Name</i>	<i>Description</i>
AddressID	[I/O] Sets/Gets current PTZ device's ID
BaudRate	[I/O] Sets/Gets PTZ baud rate
PTZPanSpeed	[I/O] Sets/Gets The PTZ pan speed
Parity	[I/O] Sets/Gets PTZ parity
Protocol	[I/O] Sets/Gets PTZ protocol
PTZFile	[I/O] Sets/Gets PTZ protocol file from local drive
PTZStatus	[O] Gets PTZ status with Enable, Disable or MousePTZ
PTZTiltSpeed	[I/O] Sets/Gets The PTZ tilt speed
Vendor	[I/O] Sets/Gets PTZ vendor

AddressID

Description

[I/O] Sets/Gets current PTZ device's ID

Syntax

AddressID = *Input_Value*
Return_Value = **AddressID**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Sets the PTZ device DIP switch ID

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current PTZ ID setting

Remarks

NONE

Example

```
objStreaming.MediaType = 4;  
.....  
objStreaming.Play() ;  
  
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
objStreaming.EnablePTZ() ;
```

See Also

BaudRate, Parity

BaudRate

Description

[I/O] Sets/Gets PTZ Baud rate.

Syntax

BaudRate = *Input_value*
Return_value = **BaudRate**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>																				
<i>Input_value</i>	LONG	Sets/Gets PTZ Baudrate. <table><tr><th>Value</th><th>Description</th></tr><tr><td>1200</td><td>Set baud rate to 1200</td></tr><tr><td>2400</td><td>Set baud rate to 2400</td></tr><tr><td>4800</td><td>Set baud rate to 4800</td></tr><tr><td>9600</td><td>Set baud rate to 9600</td></tr><tr><td>19200</td><td>Set baud rate to 19200</td></tr><tr><td>38400</td><td>Set baud rate to 38400</td></tr><tr><td>57600</td><td>Set baud rate to 57600</td></tr><tr><td>115200</td><td>Set baud rate to 115200</td></tr><tr><td>230400</td><td>Set baud rate to 230400</td></tr></table>	Value	Description	1200	Set baud rate to 1200	2400	Set baud rate to 2400	4800	Set baud rate to 4800	9600	Set baud rate to 9600	19200	Set baud rate to 19200	38400	Set baud rate to 38400	57600	Set baud rate to 57600	115200	Set baud rate to 115200	230400	Set baud rate to 230400
Value	Description																					
1200	Set baud rate to 1200																					
2400	Set baud rate to 2400																					
4800	Set baud rate to 4800																					
9600	Set baud rate to 9600																					
19200	Set baud rate to 19200																					
38400	Set baud rate to 38400																					
57600	Set baud rate to 57600																					
115200	Set baud rate to 115200																					
230400	Set baud rate to 230400																					

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns current settings value

Remarks

NONE

Example

```
objStreaming.MediaType = 4;  
.....  
objStreaming.Play() ;  
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
objStreaming.EnablePTZ() ;
```

See Also

AddressID, Parity

PTZPanSpeed

Description

[I/O] Sets/Gets PTZ pan speed

Syntax

PTZPanSpeed = *Input_Value*
Return_Value = **PTZPanSpeed**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Sets PTZ pan speed. We can set 1~5 to change pan speed.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns current settings value

Remarks

NONE

Example

```
objStreaming.MediaType = 4;  
.....  
objStreaming.Play() ;  
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
objStreaming.PTZPanSpeed = 3; // Set pan speed  
  
objStreaming.EnablePTZ() ;
```

Parity

Description

[I/O] Sets/Gets PTZ parity

Syntax

Parity = *"Input_value"*
"Return_value" = **Parity**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>								
<i>Input_value</i>	BSTR	Sets/Gets PTZ parity. <table><tr><th>Value</th><th>Description</th></tr><tr><td>N81</td><td>No parity</td></tr><tr><td>O81</td><td>Set parity to odd</td></tr><tr><td>E81</td><td>Set parity to even</td></tr></table>	Value	Description	N81	No parity	O81	Set parity to odd	E81	Set parity to even
Value	Description									
N81	No parity									
O81	Set parity to odd									
E81	Set parity to even									

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Returns current settings value

Remarks

NONE

Example

```
objStreaming.MediaType = 4;  
.....  
objStreaming.Play() ;  
  
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
objStreaming.EnablePTZ() ;
```

See Also

AddressID, BaudRate

Protocol

Description

[I/O] Sets/Gets PTZ protocol.

Syntax

```
Protocol = "Input_Value"  
"Return_Value" = Protocol
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	BSTR	Sets the ptz protocol

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Gets the ptz protocol

Remarks

NONE

Example

```
objStreaming.MediaType = 4;  
.....  
objStreaming.Play() ;  
  
objStreaming.Vendor = "NET-6100" ;  
objStreaming.Protocol = "Pelco-P" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
objStreaming.EnablePTZ() ;
```

See Also

vendor

PTZFile

Description

[I/O] Sets/Gets PTZ protocol file from local drive

Syntax

```
PTZFile = "Input_value"  
"Return_value" = PTZFile
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	BSTR	Sets the ptz protocol filename form you

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Gets the ptz protocol filename

Remarks

NONE

Example

```
objStreaming.MediaType = 4;  
.....  
objStreaming.Play() ;  
  
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
  
objStreaming.EnablePTZ() ;
```

See Also

PTZStatus

Description

[0] Gets current status of PTZ control

Syntax

Return_Value = PTZStatus [*default: 0*]

Parameters

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>										
<i>Return_Value</i>	LONG	Returns current settings value										
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Disable</td></tr><tr><td>1</td><td>Enable</td></tr><tr><td>2</td><td>MousePTZ</td></tr><tr><td>3</td><td>Absolute position</td></tr></table>	Value	Description	0	Disable	1	Enable	2	MousePTZ	3	Absolute position
Value	Description											
0	Disable											
1	Enable											
2	MousePTZ											
3	Absolute position											

Remarks

NONE

Example

```
objStreaming.MediaType = 4;
.....
objStreaming.Play() ;

objStreaming.PTZFile= "C:\MyPTZ.ptz" ;
objStreaming.AddressID = 1 ;
objStreaming.Parity = "N81" ;
objStreaming.BaudRate = 4800 ;

objStreaming.EnablePTZ() ;

nPTZStatus = objStreaming.PTZStatus ;
```

See Also

PTZTiltSpeed

Description

[I/O] Sets/Gets PTZ tilt speed

Syntax

PTZTiltSpeed = *Input_value*
Return_value = **PTZTiltSpeed**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	LONG	Sets PTZ tilt speed. We can set 1~5 to change pan speed.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns current settings value

Remarks

NONE

Example

```
objStreaming.MediaType = 4;
.....
objStreaming.Play() ;
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;
objStreaming.AddressID = 1 ;
objStreaming.Parity = "N81" ;
objStreaming.BaudRate = 4800 ;
objStreaming.PTZTiltSpeed = 3; // Set Tilt speed

objStreaming.EnablePTZ() ;
```

Vendor

Description

[I/O] Sets/Gets PTZ vendor

Syntax

vendor = *"Input_Value"*
"Return_Value" = **vendor**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	BSTR	Sets the ptz protocol vendor

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Gets the ptz protocol vendor

Remarks

NONE

Example

```
objStreaming.MediaType = 4;  
.....  
objStreaming.Play() ;  
  
objStreaming.Vendor = "NET-6100" ;  
objStreaming.Protocol = "Pelco-P" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
  
objStreaming.EnablePTZ() ;
```

See Also

Protocol

Motion Detection Property

<i>Name</i>	<i>Description</i>
MotionDetectionAlertDuration	[1/0] Sets/Gets the Duration of current MD event to show red block

MotionDetectionAlertDuration

Description

[I/O] Sets/Gets the duration of current motion detection event to show red block

Syntax

MotionDetectionAlertDuration = *Input_value*
Return_value = **MotionDetectionAlertDuration** [*default: 10*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	LONG	Sets the Duration of current MD event to show red block

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Gets the Duration of current MD event to show red block

Remarks

NONE.

Example

```
objStreaming.MediaType = 1 ;  
.....  
objStreaming.Play() ;  
  
objStreaming.MotionDetectionAlertDuration= 1 ;  
objStreaming.EnableMD() ;
```

See Also

EnableMotionDetection(), OnMDEventStart(), OnMDEventEnd(),

DIO Property

<i>Name</i>	<i>Description</i>
DigitalInputInterval	[I/O] Sets/Gets the minimum interval time to issues between each DI event

DigitalInputInterval

Description

[I/O] Sets/Gets the minimum interval time to issues between each DI events

Syntax

```
DigitalInputInterval = Input_Value  
Return_Value = DigitalInputInterval [ default: 10 ]
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	LONG	Sets the interval time between DI events

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Gets current interval time between each DI events

Remarks

When this value is set, no digital input event will be processed within this period of time.

Example

```
objStreaming.MediaType = 1 ;  
.....  
objStreaming.Play() ;  
  
objStreaming.DigitalInputInterval= 10 ;  
objStreaming.EnabledI() ;
```

See Also

onDIEvent()

3

Control Methods

This section lists media methods.

Media Methods

<i>Name</i>	<i>Description</i>
StartAudioTransfer()	Start sending audio data to streaming server
StopAudioTransfer()	Stop sending audio data to streaming server
Connect()	Connects to the media source
DecodeFrame()	Decode Frame
DecodeFrameEx()	Decode Frame
Disconnect()	Disconnects from current media source
EnableMouseMoveEvent()	Enable OnMouseMoveNotify() event.
FreeAudioToken()	Free the audio token.
GetAudioToken()	Get the audio token
GetBeginTimeString()	Get the begin time string of video.
GetEndTimeString()	Get the end time string of video.
Pause()	Pauses or resumes current action to the media stream or media file
Play()	Starts play current media stream or media file
PlayByFrame()	That can be used at Pause() frame by frame
SnapShot()	Saves current SnapShot image of current video frame by identity filename
SendAudio()	Send audio data to video server
SendURLCmdToSE()	Send URL command
SetCurrentTime()	Seek playback frame by time
SetDecodel()	Set the decoder to decode I-frame only or to decode all frames
SetPlayDirection()	Set the direction of playback forward or backward
SetPlayRate()	Set the playback speed
StartAlarmRecord()	Starts recording current video to specified file name
StartDecodeMode()	Start decode mode
StartRecord()	Start play current media stream or media file
StartStream()	Starts Decoding mpeg4 stream but don't show in the window
Stop()	Stops play current media stream or media file

StopAlarmRecord()	Stops recording current video to specified file name
StopDecodeMode	Stop decode mode.
StopRecord()	Stops recording on the current recording operation
StopPlayingAudioFile()	Stops play audio file to camera
StopStream()	Stops media streaming

StartAudioTransfer()

Description

Start sending audio data to streaming server

Syntax

LONG StartAudioTransfer()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Retuen_value</i>	LONG	Return setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

Please call **GetAudioControlToken()** in order to transmit audio from PC to the device

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

if (objStreaming.GetAudioToken()) {
    objStreaming.StartAudioTransfer() ;
}
```

See Also

stopAudioTransfer()

StopAudioTransfer()

Description

Stop sending audio data to streaming server

Syntax

```
stopAudioTransfer()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

if (objStreaming.GetAudioToken()) {
    objStreaming.StartAudioTransfer() ;
}
.....

//
objStreaming. stopAudioTransfer () ;
```

See Also

startAudioTransfer()

Connect()

Description

Connects to the media source

Syntax

```
LONG Connect( LONG AsyncConnection )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>AsyncConnection</i>	LONG	Sync/Async Connection setting <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Sync</td></tr><tr><td>1</td><td>Async</td></tr></table>	Value	Description	0	Sync	1	Async
Value	Description							
0	Sync							
1	Async							

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Retuen_value</i>	LONG	Return connection setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.Connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

See Also

Disconnect(), MediaPassword, MediaSource, MediaUsername, MulticastIP, Play()

DecodeFrame()

Description

Decode MP4 data and show it on control.

Syntax

```
LONG DecodeFrame( BYTE* pFrameData, LONG nDataLen, LONG nDataType )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>pFrameData</i>	BYTE*	The frame data of video.
<i>nDataLen</i>	LONG	The data length of video.
<i>nDataType</i>	LONG	The data type of video.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....  
objStreaming.StartDecodeMode() ;  
.....  
objStreaming.DecodeFrame( pRawBuffer, nBufferLength, nDataType ) ;  
.....  
objStreaming.StopDecodeMode() ;
```

See Also

DecodeFrameEx()

Description

Decode MP4 data and show it on control.

Syntax

LONG DecodeFrameEx(**OLE_HANDLE** *Int32BufferPointer*, **LONG** *nDataLen*, **LONG** *nDataType*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Int32BufferPointer</i>	OLE_HANDLE	The frame data of video.
<i>nDataLen</i>	LONG	The data length of video.
<i>nDataType</i>	LONG	The data type of video.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....  
objStreaming.StartDecodeMode() ;  
.....  
objStreaming.DecodeFrame( pRawBuffer, nBufferLength, nDataType ) ;  
.....  
objStreaming.StopDecodeMode() ;  
.....
```

See Also

Disconnect()

Description

Disconnects from current media source

Syntax

`Disconnect()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
  
if (ObjStreaming.NetworkStatus >=2) {  
    ObjStreaming.Disconnect() ;  
} else if (ObjStreaming.NetworkStatus ==1) {  
    ObjStreaming.DisConnControlOnly() ;  
}
```

See Also

`Connect()`

EnableMoudeMoveEvent()

Description

Enable OnMouseMoveNotify() event.

Syntax

```
EnableMoudeMoveEvent( LONG nEnable )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>nEnable</i>	Long	Enable/Disable OnMouseMoveNotify() event. Default is 0. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Disable</td></tr><tr><td>1</td><td>Enable</td></tr></table>	Value	Description	0	Disable	1	Enable
Value	Description							
0	Disable							
1	Enable							

Returns

NONE

Remarks

NONE

Example

```
objStreaming.NetworkStatus
```

See Also

[connect\(\)](#)

FreeAudioToken()

Description

Free audio token from device.

Syntax

FreeAudioToken()

Parameters

NONE

Returns

NONE

Remarks

Please call **FreeAudioToken()** after finish audio transfer or other connection can not get the token.

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

if (objStreaming.GetAudioToken()) {
    objStreaming.StartAudioTransfer() ;
}
.....
objStreaming.StopAudioTransfer() ;
objStreaming.FreeAudioToken() ;
```

See Also

GetAudioToken()

Description

Get audio token from device, only one connection can get token.

Syntax

LONG GetAudioToken()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Retuen_value</i>	LONG	Return setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

Please call **startAudioTransfer()** in order to transmit audio from PC to the device

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

if (objStreaming.GetAudioToken()) {
    objStreaming.StartAudioTransfer() ;
}
```

See Also

GetBeginTimeString()

Description

Get the begin time string of video.

Syntax

BSTR GetBeginTimeString()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Return current begin time string of video

Remarks

NONE

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
    BeginTime = objStreaming.GetBeginTimeString() ;
    EndTime = objStreaming.GetEndTimeString() ;
}
```

See Also

[GetEndTimeString\(\)](#)

GetEndTimeString()

Description

Get the End time string of video.

Syntax

BSTR GetEndTimeString()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Return current End time string of video

Remarks

NONE

Example

```
ObjStreaming.ID = 1;
ObjStreaming.MediaSource = "192.168.0.100" ;
ObjStreaming.MediaUsername = "Admin" ;
ObjStreaming.MediaPassword = "123456" ;
ObjStreaming.AutoReconnect = 1 ;
ObjStreaming.connect(0) ;
ObjStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    ObjStreaming.Play() ;
    BeginTime = ObjStreaming.GetBeginTimeString() ;
    EndTime = ObjStreaming.GetEndTimeString() ;
}
```

See Also

GetBeginTimeString()

Pause()

Description

Pauses or resumes current action to the media stream or media file

Syntax

Pause()

Parameters

NONE

Returns

This function is only valid for playback mode.

Remarks

NONE

Example

```
objStreaming.MediaType = 2 ;    'Playback mode
objStreaming.MediaSource = "C:\abc.raw" ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

objStreaming.Pause()
```

See Also

Play(), MediaType

Play()

Description

Starts play current media stream or media file

Syntax

Play()

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}
```

See Also

stop(), MediaSource, MulticastIP, Connect()

Note

When you call Play method, ActiveXControl will not call Connect and StartStream automatically. AP should call Connect and StartStream before call Play method.

PlayByFrame()

Description

Starts play current media stream or media file Step by one frame

Syntax

```
PlaybyFrame( SHORT nDirection )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>nDirection</i>	SHORT	Specifies the flag to play frame						
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Play backward</td></tr><tr><td>1</td><td>Play forward</td></tr></table>			Value	Description	0	Play backward	1	Play forward
Value	Description							
0	Play backward							
1	Play forward							

Returns

NONE

Remarks

NONE

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

objStreaming.Pause();

objStreaming.PlayByFrame(1); // Forward one frame
objStreaming.PlayByFrame(0); // Backward one frame
```

See Also

stop(), MediaSource, MulticastIP, Connect()

Snapshot()

Description

Saves current Snapshot image of current video frame by identity filename

Syntax

```
LONG Snapshot( LONG IFormat,  
               BSTR FileName,  
               LONG ReduplicateTitleBarText,  
               LONG R_TextColor,  
               LONG G_TextColor,  
               LONG B_TextColor )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>												
<i>IFormat</i>	LONG	Specifies the file type to SnapShot the video clips <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Bitmap format</td></tr><tr><td>1</td><td>JPEG format</td></tr><tr><td>2</td><td>GIF format</td></tr><tr><td>3</td><td>TIFF format</td></tr><tr><td>4</td><td>PNG format</td></tr></table>	Value	Description	0	Bitmap format	1	JPEG format	2	GIF format	3	TIFF format	4	PNG format
Value	Description													
0	Bitmap format													
1	JPEG format													
2	GIF format													
3	TIFF format													
4	PNG format													
<i>FileName</i>	BSTR	Specifies the filename to SnapShot the video clips. If filename is null string then will be copy bitmap to clipboard.												
<i>ReduplicateTitleBarText</i>	LONG	Specifies the word to the picture reduplicate.												
<i>R_TextColor</i>	LONG	Specifies the red color to the picture adjust.												
<i>G_TextColor</i>	LONG	Specifies the green color to the picture adjust.												
<i>B_TextColor</i>	LONG	Specifies the blue color to the picture adjust.												

Returns

NONE

Remarks

The function now supports bmp, jpg type of formats.

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.Snapshot(1, "c:\RecTest\Snapshot"&nFile&".jpg", 1, r_Color,
g_color, b_color) ;
}
```

See Also

[onMDSettingChange\(\)](#)

SendAudio()

Description

Saves current SnapShot image of current video frame by identity filename

Syntax

```
LONG SendAudio ( BYTE* pbyAudioBuffer, LONG nLength )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>pbyAudioBuffer</i>	BYTE*	Audio data that need to send
<i>nLength</i>	LONG	Length of audio buffer

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

The audio data will be sent by control port.

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    // Send audio data to video server
    objStreaming.SendAudio(pbyAudioBuff, lLen)
}
```

See Also

SendURLCmdToSE()

Description

Send URL command to video server.

Syntax

BSTR SendURLCmdToSE (**BSTR** *szCommand*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szCommand</i>	BSTR	URL command will be sent

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Return setting result.

Remarks

NONE

Example

```
// Send to video server, that will get result string from pbstrReturn  
objStreaming.SendURLCmdToSE("http\\192.168.1.100\\");
```

See Also

SetCurrentTime()

Description

Sets current time in playback mode

Syntax

```
SetCurrentTime ( LONG nCurrentTime )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nCurrentTime</i>	LONG	Seek position to set time

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.Play();  
objStreaming.SetCurrentTime (nCurrentTime) ;
```

See Also

SetDecodeI()

Description

Set the decoder to decode I frame only or to decode all frames

Syntax

```
SetDecodeI( SHORT DecodeIOnly )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>DecodeIOnly</i>	SHORT	Specifies the flag to decode frames						
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>All frames (default value)</td></tr><tr><td>1</td><td>I frame only</td></tr></table>	Value	Description	0	All frames (default value)	1	I frame only
Value	Description							
0	All frames (default value)							
1	I frame only							

Returns

NONE

Remarks

When application calls this function, Media Control will decode I-Frame only (without decoding P-Frame), in this way, the CPU loading will decrease dramatically

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (objStreaming.NetworkStatus != 0) {
    objStreaming.SetDecodeI(1) ;
    objStreaming.Play();
}
```

See Also

Play()

SetPlayDirection()

Description

Sets current play direction in the playback media

Syntax

```
SetPlayDirection( SHORT shDirection )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>shDirection</i>	SHORT	Specifies the flag to decode frames						
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Play Backward</td></tr><tr><td>1</td><td>Play Forward.</td></tr></table>	Value	Description	0	Play Backward	1	Play Forward.
Value	Description							
0	Play Backward							
1	Play Forward.							

Returns

NONE

Remarks

NONE

Example

```
.....

objStreaming.Play() ;
objStreaming.SetPlayDirection(1) ; // Play Forward

objStreaming.Pause() ;
objStreaming.SetPlayDirection(0) ; // Play Backward
objStreaming.Play() ;
```

See Also

SetPlayRate()

Description

Set playback speed

Syntax

SetPlayRate(LONG *nPlayRate*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>												
<i>nPlayRate</i>	LONG	Set speed of playback file												
		<table><tr><th>Value</th><th>Speed</th></tr><tr><td>0</td><td>1/2x</td></tr><tr><td>1</td><td>1x</td></tr><tr><td>2</td><td>2x</td></tr><tr><td>3</td><td>4x</td></tr><tr><td>4</td><td>8x</td></tr></table>	Value	Speed	0	1/2x	1	1x	2	2x	3	4x	4	8x
Value	Speed													
0	1/2x													
1	1x													
2	2x													
3	4x													
4	8x													

Returns

NONE

Remarks

NONE

Example

```
.....  
  
objStreaming.Play() ;  
objStreaming.SetPlayRate(2) ; // Set playback speed to 2x
```

See Also

StartAlarmRecord()

Description

Enable the alarm recording job

Syntax

```
LONG StartAlarmRecord( BSTR szFileName )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szFileName</i>	BSTR	Specifies the filename to record the video clips

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.StartAlarmRecord("C:\ABC.raw");
}
```

See Also

StartRecord(), OnStopRecord(), StopAlarmRecord(), PreRecordTime, PostRecordingTime

StartDecodeMode ()

Description

Starts decoding current video to media mode.

Syntax

LONG StartDecodeMode()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.StartDecodeMode() ;
.....
objStreaming.DecodeFrame( pRawBuffer, nBufferLength, nDataType ) ;
.....
objStreaming.StopDecodeMode() ;
```

See Also

StartRecord()

Description

Starts recording current video to specified file name

Syntax

```
LONG StartRecord( BSTR szFileName )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szFileName</i>	BSTR	Specifies the filename to record the video clips

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
BOOL bRecording = false;
ObjStreaming.ID = 1;
ObjStreaming.MediaSource = "192.168.0.100" ;
ObjStreaming.MediaUsername = "Admin" ;
ObjStreaming.MediaPassword = "123456" ;
ObjStreaming.AutoReconnect = 1 ;
ObjStreaming.connect(0) ;
ObjStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    ObjStreaming.StartRecord("C:\ABC.raw");
    bRecording = true;
}
If(bRecording){
    ObjStreaming.StopRecord();
```

```
}  
}
```

See Also

`StopRecord()`

StartStream()

Description

Starts Decoding mpeg4 stream but don't show in the window

Syntax

startStream()

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
objStreaming.ID = 1;
objStreaming.MediaSource = "192.168.0.100" ;
objStreaming.MediaUsername = "Admin" ;
objStreaming.MediaPassword = "123456" ;
objStreaming.AutoReconnect = 1 ;
objStreaming.connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.StartStream();
}
```

See Also

stopRecord()

Stop()

Description

Stops play current media stream or media file

Syntax

stop()

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....
objStreaming.connect(0) ;
objStreaming.StartStream();

If (ObjStreaming.NetworkStatus != 0) {
    objStreaming.Play() ;
}

.....

If (ObjStreaming.ContentStatus == 1) {
    objStreaming.Stop() ;
}
```

See Also

Play()

StopAlarmRecord()

Description

Stops alarm recording job

Syntax

```
stopAlarmRecord()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
  
if (objStreaming.RecordType == 2) {  
    objStreaming.StopAlarmRecord() ;  
} else if (objStreaming.RecordType == 1) {  
    objStreaming.StopRecord() ;  
}
```

See Also

[StartRecord\(\)](#), [OnStopRecord\(\)](#), [StartAlarmRecord\(\)](#), [PreRecordTime](#),
[PostRecordingTime](#)

StopDecodeMode()

Description

Stops decoding current video to media mode

Syntax

```
stopDecodeMode()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.StartDecodeMode() ;  
.....  
objStreaming.DecodeFrame( pRawBuffer, nBufferLength, nDataType ) ;  
.....  
objStreaming.StopDecodeMode() ;
```

See Also

startDecodeMode ()

StopRecord()

Description

Stops recording on the current recording operation

Syntax

StopRecord()

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
  
if (objStreaming.RecordType == 2) {  
    objStreaming.StopAlarmRecord() ;  
} else if (objStreaming.RecordType == 1) {  
    objStreaming.StopRecord() ;  
}
```

See Also

startRecord(), **OnStopRecord()**

StopPlayingAudioFile()

Description

Stop playing audio file to camera.

Syntax

`StopPlayingAudioFile()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.MediaType = 6  
.....  
objStreaming.PlayAudioFileToDevice("C:\Audio.wav")  
.....  
objStreaming.StopPlayingAudioFile()
```

See Also

`PlayAudioFileToDevice()`

StopStream()

Description

Stop Decoding mpeg4 stream

Syntax

`stopStream()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.connect(0) ;  
  
If (ObjStreaming.NetworkStatus != 0) {  
    objStreaming.StartStream() ;  
}  
  
.....  
  
If (ObjStreaming.ContentStatus == 1) {  
    objStreaming.StopStream() ;  
}
```

See Also

`StopRecord()`

SetVideoTransformConfig()

Description

Set the video transform config to force streaming engine resize video.

Syntax

```
SetVideoTransFormConfig( USHORT bTransfer,  
                          LONG nVideoOutType,  
                          LONG BitRate,  
                          LONG nFpsNum,  
                          USHORT bReSize,  
                          LONG nResolution )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>bTransfer</i>	USHORT	0 is false 1 is true
<i>nVideoOutType</i>	LONG	1 : Mpeg4 4 : MJPEG 5 : H.264
<i>BitRate</i>	LONG	Ex:500000 is 500kbps
<i>nFpsNum</i>	LONG	Frame rate
<i>bReSize</i>	USHORT	0 is disable, 1 is enable.
<i>nResolution</i>	LONG	Refer to Resolution map

Returns

NONE

Remarks

See Also

UI Methods

<i>Name</i>	<i>Description</i>
DisplayTitleBar()	Show or hide title bar of the media window
EnableFullScreen()	Enable the FullScreen
SetControlActive()	Activate or de-activate current media window
SetDefaultBackgroundColor ()	Set default background color
SetTitleBarTextLayout()	Set the title bar text layout
SetTitleActiveColor()	Set the title color in active mode.
SetTitleNonActiveColor()	Set the title color in Non-active mode.
SetTitleEventColor()	Set the title color in event mode
SetPlayingBackgroundColor()	Set the Playing background color
SetBorderWidth	Set the control border width.
SetBorderColor	Set the control border color.

DisplayTitleBar()

Description

Show or hide title bar of the media window

Syntax

```
DisplayTitleBar( LONG Display )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Display</i>	LONG	Show or hide title bar of the media window

Value	Description
0	Hide the title bar of the media window
1	Display the title bar of the media window

Returns

NONE

Remarks



NOTE: This function is useful when you want to develop a multiple-channel preview or playback, e.g. 16-channel preview, and the preview window is too small to accommodate any title bar

Example

```
'---- display title bar  
objStreaming.DisplayTitleBar(1) ;
```

See Also

EnableFullScreen()

Description

Enable the FullScreen.

Syntax

EnableFullScreen()

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
objStreaming.EnableFullScreen () ;
```

See Also

SetControlActive()

Description

Activate or de-activate current media window

Syntax

`SetControlActive(LONG nActive)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>nActive</i>	LONG	Activate or de-activate current media window						
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Set the window to de-active</td></tr><tr><td>1</td><td>Set the window to Active</td></tr></table>	Value	Description	0	Set the window to de-active	1	Set the window to Active
Value	Description							
0	Set the window to de-active							
1	Set the window to Active							

Returns

NONE

Remarks

NONE

Example

```
objStreaming.SetControlActive (1) ;
```

See Also

SetDefaultBackgroundColor ()

Description

Set the default background color.

Syntax

```
setDefaultBackgroundColor( LONG R, LONG G, LONG B )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>R</i>	LONG	0~255
<i>G</i>	LONG	0~255
<i>B</i>	LONG	0~255

Returns

NONE

Remarks

NONE

Example

```
objStreaming.SetDefaultBackgroundColor (255, 255, 255) ;
```

See Also

SetTitleBarTextLayout()

Description

Set the title bar text layout

Syntax

```
SetTitleBarTextLayout( LONG n1,  
                      LONG n2,  
                      LONG n3,  
                      LONG n4,  
                      LONG n5,  
                      LONG n6,  
                      LONG n7,  
                      LONG n8,  
                      LONG n9,  
                      LONG n10,  
                      LONG n11,  
                      LONG n12,  
                      LONG n13,  
                      LONG n14,  
                      LONG n15,  
                      LONG n16,  
                      LONG n17,  
                      LONG n18,  
                      LONG n19,  
                      LONG n20 );
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>n1</i>	LONG	
<i>n2</i>	LONG	
<i>n3</i>	LONG	
<i>n4</i>	LONG	
<i>n5</i>	LONG	
<i>n6</i>	LONG	
<i>n7</i>	LONG	
<i>n8</i>	LONG	
<i>n9</i>	LONG	
<i>n10</i>	LONG	

<i>n11</i>	LONG
<i>n12</i>	LONG
<i>n13</i>	LONG
<i>n14</i>	LONG
<i>n15</i>	LONG
<i>n16</i>	LONG
<i>n17</i>	LONG
<i>n18</i>	LONG
<i>n19</i>	LONG
<i>n20</i>	LONG

Returns

NONE

Remarks

NONE

Example

```
objStreaming.SetTitleBarTextLayout(9, 12, 10, 19,
1, 14, 2, 14, 3, 18, 5, 12, 6, 12, 7, 0, 0, 0, 0,
0) ;
```

See Also

Value	Description
0	NONE
1	YYYY EX:2008
2	YY EX:08
3	MM EX:12
4	DD EX:28
5	hh EX:23
6	mm EX:29
7	ss EX:59
8	weekday EX:"Mon"
9	ID EX:1
10	Caption EX:"Front Door"
11	Slash EX: "/"
12	Colon EX: "."
13	At EX: "@"
14	Dash EX: "-"
15	LeftBracket EX: "["
16	RightBracket EX: "]"
17	Space EX: " "
18	Space 3 EX: " "
19	Space 5 EX: " "
20	FPS
21	ms

SetTitleActiveColor()

Description

Set the title color in active mode. Title is consisting of 16 lines.

Syntax

```
SetTitleActiveColor( LONG nIndex, LONG R, LONG G, LONG B )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nIndex</i>	LONG	Line number is range from 0~15.
<i>R</i>	LONG	0~255
<i>G</i>	LONG	0~255
<i>B</i>	LONG	0~255

Returns

NONE

Remarks

NONE

Example

```
objStreaming.SetTitleActiveColor(1, 255, 255, 255) ;
```

See Also

SetTitleNonActiveColor()

Description

Set the title color in non-active mode. Title is consisting of 16 lines

Syntax

```
SetTitleNonActiveColor( LONG nIndex, LONG R, LONG G, LONG B)
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nIndex</i>	LONG	Line number is range from 0~15.
<i>R</i>	LONG	0~255
<i>G</i>	LONG	0~255
<i>B</i>	LONG	0~255

Returns

NONE

Remarks

NONE

Example

```
objStreaming.SetTitleNonActiveColor(1, 255, 255, 255) ;
```

See Also

SetTitleEventColor()

Description

Set the title color in event mode. Title is consist of 16 lines

Syntax

```
SetTitleEventColor( LONG nIndex, LONG R, LONG G, LONG B)
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nIndex</i>	LONG	Line number is range from 0~15.
<i>R</i>	LONG	0~255
<i>G</i>	LONG	0~255
<i>B</i>	LONG	0~255

Returns

NONE

Remarks

NONE

Example

```
objstreaming.SetTitleEventColor(1, 255, 255, 255) ;
```

See Also

SetPlayingBackgroundColor()

Description

Set the playing background color

Syntax

```
SetPlayingBackgroundColor( LONG R, LONG G, LONG B )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>R</i>	LONG	0~255
<i>G</i>	LONG	0~255
<i>B</i>	LONG	0~255

Returns

NONE

Remarks

NONE

Example

```
objStreaming.SetPlayingBackgroundColor (255, 255, 255) ;
```

See Also

SetBorderWidth()

Description

Set the control border width.

Syntax

```
SetBorderWidth( LONG value )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>value</i>	LONG	Specifies the width of the pen, in logical units.

Returns

NONE

Remarks

See Also

SetBorderColor()

Description

Set the control border color.

Syntax

```
SetBorderColor( LONG colorR, LONG colorG, LONG colorB )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>colorR</i>	LONG	Specifies the intensity of the red color.
<i>colorG</i>	LONG	Specifies the intensity of the green color.
<i>colorB</i>	LONG	Specifies the intensity of the blue color.

Returns

NONE

Remarks

See Also

PTZ Methods

<i>Name</i>	<i>Description</i>
DigitalZoomIn()	Operate the Digital Zoom In functions
DigitalZoomOut()	Operate the Digital Zoom Out functions
DigitalMoveLeft ()	Operate the Digital Move Left functions
DigitalMoveRight ()	Operate the Digital Move Right functions
DigitalMoveUp ()	Operate the Digital Move Up functions
DigitalMoveDown ()	Operate the Digital Move Down functions
DigitalZoomOutMax ()	Operate the Digital Zoom Out Max functions
DisableMouseDigitalPTZ()	Disable Mouse Digital PTZ function
DisableMousePTZ()	Disable Mouse PTZ functions
DisablePTZ()	Disable PTZ functions
EnableMousePTZ()	Enable Mouse PTZ functions
EnableMouseDigitalPTZ()	Enable Mouse Digital PTZ function
EnablePTZ()	Enable PTZ functions
EnumerateProtocol()	Enumerate available protocol
PTZBLC()	Operate the camera BLC function
PTZDayNight()	Operate the camera day&night function
PTZFocus()	Operate the camera focus function
PTZIris()	Operate the camera iris function
PTZMove()	Move the camera view
PTZOSD()	Operate the camera OSD function
PTZPreset()	Operate the camera preset functions
PTZZoom()	Operate the camera Zoom functions
SendPTZCmd()	Send PTZ protocol command to video server or speed dome

DigitalZoomIn()

Description

Operate the Digital Zoom In functions

Syntax

LONG DigitalZoomIn(**LONG** *nSteps*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nSteps</i>	LONG	Set the special the digital zoom in value.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.DigitalZoomIn(LONG nSteps)
```

See Also

PTZZoom()

DigitalZoomOut()

Description

Operate the Digital Zoom In functions

Syntax

LONG DigitalZoomOut(**LONG** *nSteps*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nSteps</i>	LONG	Set the special the digital zoom Out value.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.DigitalZoomOut(LONG nSteps)
```

See Also

PTZZoom()

DigitalMoveLeft ()

Description

Operate the Digital Zoom In functions

Syntax

LONG DigitalMoveLeft(**LONG** *nSteps*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nSteps</i>	LONG	Set the special the digital move left value.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.DigitalMoveLeft(LONG nSteps);
```

See Also

PTZZoom()

DigitalMoveRight ()

Description

Operate the Digital Zoom In functions

Syntax

LONG DigitalMoveRight(**LONG** *nSteps*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nSteps</i>	LONG	Set the special the digital move right value.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.DigitalMoveRight(LONG nSteps);
```

See Also

PTZZoom()

DigitalMoveUp ()

Description

Operate the Digital Zoom In functions

Syntax

LONG DigitalMoveUp(**LONG** *nSteps*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nSteps</i>	LONG	Set the special the digital move Up value.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.DigitalMoveUp(LONG nSteps);
```

See Also

PTZZoom()

DigitalMoveDown ()

Description

Operate the Digital Zoom In functions

Syntax

LONG DigitalMoveDown(**LONG** *nSteps*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nSteps</i>	LONG	Set the special the digital move down value.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.DigitalMoveDown(LONG nSteps);
```

See Also

PTZZoom()

DigitalZoomOutMax ()

Description

Operate the Digital Zoom In functions

Syntax

DigitalZoomOutMax ()

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
objStreaming.DigitalZoomOutMax();
```

See Also

PTZZoom()

DisableMouseDigitalPTZ()

Description

Disable Mouse Digital PTZ functions

Syntax

```
DisableMouseDigitalPTZ()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
  
objStreaming.EnablePTZ() ;  
objStreaming.EnableMouseDigitalPTZ() ;  
  
objStreaming.DisableMouseDigitalPTZ() ;  
objStreaming.DisablePTZ() ;
```

See Also

EnableMouseDigitalPTZ()

DisableMousePTZ()

Description

Disable Mouse PTZ functions

Syntax

`DisableMousePTZ()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
  
objStreaming.EnablePTZ() ;  
objStreaming.EnableMousePTZ() ;  
  
objStreaming.DisableMousePTZ() ;  
objStreaming.DisablePTZ() ;
```

See Also

`EnableMousePTZ()`

DisablePTZ()

Description

Disable PTZ functions

Syntax

`DisablePTZ()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
  
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
  
objStreaming.EnablePTZ() ;  
  
objStreaming.DisablePTZ() ;
```

See Also

`EnablePTZ()`

EnableMousePTZ()

Description

Enable Mouse PTZ functions

Syntax

`LONG EnableMousePTZ()`

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;
objStreaming.AddressID = 1 ;
objStreaming.Parity = "N81" ;
objStreaming.BaudRate = 4800 ;

objStreaming.EnablePTZ() ;
objStreaming.EnableMousePTZ() ;

objStreaming.DisableMousePTZ() ;
objStreaming.DisablePTZ() ;
```

See Also

`DisableMousePTZ()`

EnableMouseDigitalPTZ()

Description

Enable Mouse Digital PTZ functions

Syntax

`EnableMouseDigitalPTZ()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
objStreaming.EnableMouseDigitalPTZ() ;  
  
objStreaming.DisableMouseDigitalPTZ() ;
```

See Also

`DisableMouseDigitalPTZ()`

EnablePTZ()

Description

Enable PTZ functions

Syntax

LONG EnablePTZ()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result.						
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>			Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;
objStreaming.AddressID = 1 ;
objStreaming.Parity = "N81" ;
objStreaming.BaudRate = 4800 ;

objStreaming.EnablePTZ() ;

objStreaming.DisablePTZ() ;
```

See Also

DisablePTZ()

EnumerateVendor

Description

Enumerate available Vendor that was changed from method to property.

Syntax

BSTR EnumerateVendor()

Remarks

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Available vendor list

Remarks

We can use the property to get vendor that was supported and it was formatted as first number is how many vendors were supported. Every vendor was separate with ‘,’. For example, 3,CAM-6100,CAM-6200,CAM-6500. First number 3 means there have 3 vendor fellows.

Example

```
strVendor = objStreaming.EnumerateVendor(); // Get available vendor

objStreaming.PTZFile= "C:\MyPTZ.ptz" ;
objStreaming.AddressID = 1 ;
objStreaming.Parity = "N81" ;
objStreaming.BaudRate = 4800 ;

objStreaming.EnablePTZ() ;
```

See Also

EnumerateProtocol()

EnumerateProtocol()

Description

Enumerate available protocol

Syntax

```
BSTR EnumerateProtocol( BSTR szVendor )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szVendor</i>	BSTR	Specifies the vendor

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Returns the protocol listing of the vendor specified

Remarks

NONE

Example

```
objStreaming.PTZFile= "C:\MyPTZ.ptz" ;  
objStreaming.AddressID = 1 ;  
objStreaming.Parity = "N81" ;  
objStreaming.BaudRate = 4800 ;  
  
objStreaming.EnablePTZ() ;  
szProtocol = objStreaming.EnumerateProtocol("NET-6100");
```

See Also

PTZBLC()

Description

Operate the camera BLC function

Syntax

PTZBLC(BSTR *szInputCommand*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>szInputCommand</i>	BSTR	Move the camera view <table><tr><th>Value</th><th>Description</th></tr><tr><td>ON</td><td>BLC on</td></tr><tr><td>OFF</td><td>BLC off</td></tr></table>	Value	Description	ON	BLC on	OFF	BLC off
Value	Description							
ON	BLC on							
OFF	BLC off							

Returns

NONE

Remarks

NONE

Example

```
.....
objStreaming.EnablePTZ() ;
objStreaming.PTZSetPanSpeed(3) ;
objStreaming.PTZSetTiltSpeed(3) ;
objStreaming. PTZBLC("ON") ;
//
objStreaming.PTZBLC("OFF") ;
```

See Also

PTZDayNight()

Description

Operate the camera Day and Night function

Syntax

PTZDayNight(BSTR *szInputCommand*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>										
<i>szInputCommand</i>	BSTR	Move the camera view <table><tr><th>Value</th><th>Description</th></tr><tr><td>On</td><td>Day and night function on</td></tr><tr><td>OFF</td><td>Day and night function off</td></tr><tr><td>AUTOON</td><td>Day and night function auto on</td></tr><tr><td>AUTOOFF</td><td>Day and night function auto off</td></tr></table>	Value	Description	On	Day and night function on	OFF	Day and night function off	AUTOON	Day and night function auto on	AUTOOFF	Day and night function auto off
Value	Description											
On	Day and night function on											
OFF	Day and night function off											
AUTOON	Day and night function auto on											
AUTOOFF	Day and night function auto off											

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.EnablePTZ() ;  
objStreaming.PTZSetPanSpeed(3) ;  
objStreaming.PTZSetTiltSpeed(3) ;  
objStreaming.PTZDayNight("ON") ;  
//  
objStreaming.PTZDayNight("OFF") ;
```

See Also

PTZFocus()

Description

Operate the camera Focus function

Syntax

PTZFocus(BSTR *szInputCommand*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>								
<i>szInputCommand</i>	BSTR	Move the camera view <table><tr><th>Value</th><th>Description</th></tr><tr><td>IN</td><td>Focus In</td></tr><tr><td>OUT</td><td>Focus Out</td></tr><tr><td>STOP</td><td>Stop Focus Action</td></tr></table>	Value	Description	IN	Focus In	OUT	Focus Out	STOP	Stop Focus Action
Value	Description									
IN	Focus In									
OUT	Focus Out									
STOP	Stop Focus Action									

Returns

NONE

Remarks

NONE

Example

```
.....
objStreaming.EnablePTZ() ;
objStreaming.PTZSetPanSpeed(3) ;
objStreaming.PTZSetTiltSpeed(3) ;
objStreaming.PTZDayNight("ON") ;
//
objStreaming.PTZDayNight("OFF") ;
```

See Also

PTZIris()

Description

Operate the camera iris function

Syntax

```
PTZIris( BSTR szInputCommand )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>										
<i>szInputCommand</i>	BSTR	Move the camera view <table><tr><th>Value</th><th>Description</th></tr><tr><td>OPEN</td><td>Open iris</td></tr><tr><td>CLOSE</td><td>Close iris</td></tr><tr><td>STOP</td><td>Stop iris</td></tr><tr><td>AUTO</td><td>Open auto iris</td></tr></table>	Value	Description	OPEN	Open iris	CLOSE	Close iris	STOP	Stop iris	AUTO	Open auto iris
Value	Description											
OPEN	Open iris											
CLOSE	Close iris											
STOP	Stop iris											
AUTO	Open auto iris											

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.EnablePTZ() ;  
objStreaming.PTZSetPanSpeed(3) ;  
objStreaming.PTZSetTiltSpeed(3) ;  
objStreaming.PTZIris("OPEN") ;  
//  
objStreaming.PTZIRIS("CLOSE") ;
```

See Also

PTZMove()

Description

Move the camera view

Syntax

PTZMove(BSTR szInputCommand)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>																						
<i>szInputCommand</i>	BSTR	Move the camera view <table><tr><th>Value</th><th>Description</th></tr><tr><td>LEFT</td><td>Pan Left</td></tr><tr><td>RIGHT</td><td>Pan Right</td></tr><tr><td>UP</td><td>Tilt Up</td></tr><tr><td>DOWN</td><td>Tilt Down</td></tr><tr><td>STOP</td><td>Stop moving</td></tr><tr><td>HOME</td><td>Back to home position</td></tr><tr><td>UPLEFT</td><td>Pan left and tilt up</td></tr><tr><td>UPRIGHT</td><td>Pan right and tilt up</td></tr><tr><td>DOWNLEFT</td><td>Pan left and tilt down</td></tr><tr><td>DOWNRIGHT</td><td>Pan right and tilt down</td></tr></table>	Value	Description	LEFT	Pan Left	RIGHT	Pan Right	UP	Tilt Up	DOWN	Tilt Down	STOP	Stop moving	HOME	Back to home position	UPLEFT	Pan left and tilt up	UPRIGHT	Pan right and tilt up	DOWNLEFT	Pan left and tilt down	DOWNRIGHT	Pan right and tilt down
Value	Description																							
LEFT	Pan Left																							
RIGHT	Pan Right																							
UP	Tilt Up																							
DOWN	Tilt Down																							
STOP	Stop moving																							
HOME	Back to home position																							
UPLEFT	Pan left and tilt up																							
UPRIGHT	Pan right and tilt up																							
DOWNLEFT	Pan left and tilt down																							
DOWNRIGHT	Pan right and tilt down																							

Returns

NONE

Remarks

NONE

Example

```
.....
objStreaming.EnablePTZ() ;
objStreaming.PTZSetPanSpeed(3) ;
objStreaming.PTZSetTiltSpeed(3) ;
objStreaming.PTZMOVE("LEFT") ;
//
objStreaming.PTZMOVE("RIGHT") ;
//
objStreaming.PTZMOVE("UP") ;
//
objStreaming.PTZMOVE("DOWN") ;
//
```



```
objStreaming.PTZMOVE("UPRIGHT") ;  
objStreaming.PTZMOVE("UPLEFT") ;  
objStreaming.PTZMOVE("DOWNRIGHT") ;  
objStreaming.PTZMOVE("DOWNLEFT") ;  
//  
objStreaming.PTZMOVE("STOP") ;  
//  
objStreaming.PTZMOVE("HOME") ;
```

See Also

PTZOSD()

Description

Operate the camera OSD function

Syntax

PTZOSD(BSTR *szInputCommand*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>																		
<i>szInputCommand</i>	BSTR	Move the camera view <table><tr><th>Value</th><th>Description</th></tr><tr><td>ON</td><td>Open OSD</td></tr><tr><td>OFF</td><td>Close OSD</td></tr><tr><td>LEFT</td><td>OSD Cursor move left</td></tr><tr><td>RIGHT</td><td>OSD Cursor move right</td></tr><tr><td>UP</td><td>OSD Cursor move up</td></tr><tr><td>DOWN</td><td>OSD Cursor move down</td></tr><tr><td>ENTER</td><td>Enter OSD sub menu</td></tr><tr><td>LEAVE</td><td>Leave OSD menu</td></tr></table>	Value	Description	ON	Open OSD	OFF	Close OSD	LEFT	OSD Cursor move left	RIGHT	OSD Cursor move right	UP	OSD Cursor move up	DOWN	OSD Cursor move down	ENTER	Enter OSD sub menu	LEAVE	Leave OSD menu
Value	Description																			
ON	Open OSD																			
OFF	Close OSD																			
LEFT	OSD Cursor move left																			
RIGHT	OSD Cursor move right																			
UP	OSD Cursor move up																			
DOWN	OSD Cursor move down																			
ENTER	Enter OSD sub menu																			
LEAVE	Leave OSD menu																			

Returns

NONE

Remarks

NONE

Example

```
.....
objStreaming.EnablePTZ() ;
objStreaming.PTZSetPanSpeed(3) ;
objStreaming.PTZSetTiltSpeed(3) ;
objStreaming.PTZOSD("ON") ;
//
objStreaming.PTZOSD("OFF") ;
```

See Also

PTZPreset()

Description

Operate the camera preset functions

Syntax

PTZPreset(BSTR *szInputCommand*, LONG *nPresetIndex*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>								
<i>szInputCommand</i>	BSTR	Move the camera view <table><tr><th>Value</th><th>Description</th></tr><tr><td>SET</td><td>Set the preset position</td></tr><tr><td>CLEAR</td><td>Clear the preset position</td></tr><tr><td>GOTO</td><td>Goto the preset position</td></tr></table>	Value	Description	SET	Set the preset position	CLEAR	Clear the preset position	GOTO	Goto the preset position
Value	Description									
SET	Set the preset position									
CLEAR	Clear the preset position									
GOTO	Goto the preset position									
<i>nPresetIndex</i>	LONG	Move the camera view <table><tr><th>Minimum</th><th>Maximum</th></tr><tr><td>1</td><td>16</td></tr></table>	Minimum	Maximum	1	16				
Minimum	Maximum									
1	16									

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.EnablePTZ() ;  
objStreaming.PTZSetPanSpeed(3) ;  
objStreaming.PTZSetTiltSpeed(3) ;  
objStreaming. PTZPreset("SET", 1) ;  
//  
objStreaming. PTZPreset("CLEAR", 1) ;  
//  
objStreaming. PTZPreset("GOTO", 1) ;
```

See Also

PTZZoom()

Description

Operate the camera Zoom functions

Syntax

PTZZoom(BSTR szInputCommand)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>								
<i>szInputCommand</i>	BSTR	Move the camera view <table><tr><th>Value</th><th>Description</th></tr><tr><td>IN</td><td>Zoom in</td></tr><tr><td>OUT</td><td>Zoom out</td></tr><tr><td>STOP</td><td>Stop moveing</td></tr></table>	Value	Description	IN	Zoom in	OUT	Zoom out	STOP	Stop moveing
Value	Description									
IN	Zoom in									
OUT	Zoom out									
STOP	Stop moveing									

Returns

NONE

Remarks

NONE

Example

```
.....
objStreaming.EnablePTZ() ;
objStreaming.PTZSetPanSpeed(3) ;
objStreaming.PTZSetTiltSpeed(3) ;
objStreaming.PTZZOOM("IN") ;
//
objStreaming.PTZZOOM("OUT") ;
//
objStreaming.PTZSTOP("STOP") ;
```

See Also

SendPTZCmd()

Description

Send ASCII string command to video server

Syntax

```
LONG SendPTZCmd( BSTR szCommand, LONG nParam1, LONG nParam2 )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szCommand</i>	BSTR	The PTZ command that is defined by protocol file
<i>nParam1</i>	SHORT	The Param1 that is defined by protocol file
<i>nParam2</i>	SHORT	The Param2 that is defined by protoco2 file

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>								
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr><tr><td>2</td><td>Else</td></tr></table>	Value	Description	0	Fail	1	Success	2	Else
Value	Description									
0	Fail									
1	Success									
2	Else									

Remarks

NONE

Example

```
objPTZ.SendPTZCmd("PANLEFT", 3, 0); // Send a PANLEFT command, speed 3 to speed dome.
```

See Also

Motion Detection Methods

<i>Name</i>	<i>Description</i>
DisableMotionDetection()	Disable or delete motion detection area
EnableMotionDetection()	Enable or add new motion detection area
GetMotionSensitive()	Get motion sensitive value.
GetMotionStartX()	Get motion start X position
GetMotionStartY()	Get motion start Y position
GetMotionEndX()	Get motion end X position
GetMotionEndY()	Get motion end Y position
SetMotionSetting()	Set motion detection settings
SetQuadMotionDetection()	Set quad motion detection.
StartMDSetup()	Set the motion status to SETTING mode
StopMDSetup()	Set the motion status to normal mode
SyncMDInfoFromDevice()	Synchronize motion detect setting value on ip device

DisableMotionDetection()

Description

Disable motion detection

Syntax

`DisableMotionDetection()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....

objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//
objStreaming.DisableMotionDetection () ;
```

See Also

`EnableMotionDetection()`

EnableMotionDetection()

Description

Enable motion detection

Syntax

`LONG EnableMotionDetection()`

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....

objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//
objStreaming.DisableMotionDetection () ;
```

See Also

`DisableMotionDetection()`, `OnMDEventStart()`

GetMotionSensitive()

Description

Get motion sensitive value

Syntax

```
LONG GetMotionSensitive( LONG nRegionNumber )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nRegionNumber</i>	LONG	Set the special the region number.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....

objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;
MotionSensitive = objStreaming.GetMotionSensitive (100) ;

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//
objStreaming.DisableMotionDetection () ;
```

See Also

setMotionSensitivity()

GetMotionStartX()

Description

Get motion start X position

Syntax

LONG GetMotionStartX(**LONG** *nRegionNumber*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nRegionNumber</i>	LONG	Set the special the region number.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....

objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;
StartX = objStreaming.GetMotionStartX (100);

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//
objStreaming.DisableMotionDetection () ;
```

See Also

GetMotionStartY()

Description

Get motion start Y position

Syntax

```
LONG GetMotionStartY( LONG nRegionNumber )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nRegionNumber</i>	LONG	Set the special the region number.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....

objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;
startX = GetMotionStartY (100);

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//
objStreaming.DisableMotionDetection () ;
```

See Also

GetMotionEndX()

Description

Get motion end X position

Syntax

```
LONG GetMotionEndX( LONG nRegionNumber )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nRegionNumber</i>	LONG	Set the special the region number.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....

objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;
EndX = GetMotionEndX (100);

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//
objStreaming.DisableMotionDetection () ;
```

See Also

GetMotionEndY()

Description

Get motion end Y position

Syntax

```
LONG GetMotionEndY( LONG nRegionNumber )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nRegionNumber</i>	LONG	Set the special the region number.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....

objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;
EndY = GetMotionEndY (100);

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//
objStreaming.DisableMotionDetection () ;
```

See Also

GetMotionRegionEnable()

Description

Get motion sensitive value

Syntax

```
ULONG GetMotionRegionEnable( LONG nRegionNumber )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nRegionNumber</i>	LONG	Set the special the region number.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	ULONG	Returns current motion enable status.

Remarks

NONE

SetMotionRegionEnable()

Description

Set motion enable status

Syntax

```
SetMotionRegionEnable( LONG nRegionNumber, ULONG u1Enable )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nRegionNumber</i>	LONG	Motion detection window number. Valid values are: 1, 2, 3
<i>u1Enable</i>	ULONG	Enable status

Returns

NONE

Remarks

NONE

SetMotionSetting()

Description

Set motion detection settings

Syntax

```
SetMotionSetting( LONG nMotionRegionNumber,  
                  LONG StartX,  
                  LONG StartY,  
                  LONG EndX,  
                  LONG EndY,  
                  LONG nSesitive )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nMotionRegionNumber</i>	LONG	Motion detection window number. Valid values are: 1, 2, 3
<i>StartX</i>	LONG	Motion detection window region starting X value
<i>StartY</i>	LONG	Motion detection window region starting Y value
<i>EndX</i>	LONG	Motion detection window region ending X value
<i>EndY</i>	LONG	Motion detection window region ending Y value
<i>nSesitive</i>	LONG	Motion detection window region sensitivity

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.StartMDSetup() ;  
  
objStreaming.SetMotionSetting(1,5,5,100,100,50) ;  
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;  
objStreaming.SetMotionSensitivity(1,80) ;  
  
objStreaming.StopMDSetup() ;  
  
objStreaming.EnableMotionDetection () ;  
//---- motion detection callback will be passed  
objStreaming.DisableMotionDetection() ;
```

See Also

`DisableMotionDetection()`, `OnMDEventStart()`

SetQuadMotionDetection()

Description

Set Quad Motion Detection

Syntax

```
LONG SetQuadMotionDetection( BYTE byEnable )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>byEnable</i>	BYTE	Set quad motion detection number.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....
objStreaming.StartMDSetup() ;

objStreaming.SetMotionSensitivity(1,80) ;
QuadMotionDetection = objStreaming.SetQuadMotionDetection(1);

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//---- motion detection callback will be passed
objStreaming.DisableMotionDetection() ;
```

See Also

SetMotionSensitivity()

SetMotionSensitivity()

Description

Set motion detection settings

Syntax

```
SetMotionSensitivity( LONG nMotionRegionNumber, LONG nSensitive )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nMotionRegionNumber</i>	LONG	Motion detection window number. Valid values are: 1, 2, 3
<i>nSensitive</i>	LONG	Motion detection window region starting X value

Returns

NONE

Remarks

NONE

Example

```
.....
objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//---- motion detection callback will be passed
objStreaming.DisableMotionDetection() ;
```

See Also

DisableMotionDetection(), OnMDEventStart()

StartMDSetup()

Description

Set the motion status to setting mode

Syntax

LONG StartMDSetup()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....

objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//---- motion detection callback will be passed
objStreaming.DisableMotionDetection() ;
```

See Also

DisableMotionDetection(), OnMDEventStart()

StopMDSetup()

Description

Set the motion status to normal mode

Syntax

```
stopMDSetup()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....

objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//---- motion detection callback will be passed
objStreaming.DisableMotionDetection() ;
```

See Also

`DisableMotionDetection()`, `OnMDEventStart()`

SyncMDInfoFromDevice()

Description

Synchronize motion detect setting value on ip device.

Syntax

LONG SyncCMDInfoFromDevice()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

Synchronize settings in first time.

Example

```
.....
SyncMDInfoFromDevice() ;
objStreaming.StartMDSetup() ;

objStreaming.SetMotionSetting(1,5,5,100,100,50) ;
objStreaming.SetMotionSetting(2,105,105,150,150,30) ;
objStreaming.SetMotionSensitivity(1,80) ;

objStreaming.StopMDSetup() ;

objStreaming.EnableMotionDetection () ;
//---- motion detection callback will be passed
objStreaming.DisableMotionDetection() ;
```

See Also

DisableMotionDetection(), OnMDEventStart()

DIO Methods

<i>Name</i>	<i>Description</i>
DisableDigitalInput()	Enable digital input that will receive data from video server
EnableDigitalInput()	Disable digital input that will not receive data from video server
DigitalOutput	Trigger assigned digital out event

DisableDigitalInput()

Description

Disable to receive Digital input event

Syntax

`DisableDigitalInput()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
If (ObjStreaming.NetworkStatus != 0) {  
    objStreaming.StartStream();  
}  
  
objStreaming.EnableDigitalInput () ;  
//  
objStreaming.DisableDigitalInput () ;
```

See Also

`EnableMotionDetection()`

EnableDigitalInput()

Description

Enable and receive digital input event

Syntax

LONG EnableDigitalInput()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Return one.

Remarks

NONE

Example

```
.....  
If (ObjStreaming.NetworkStatus != 0) {  
    objStreaming.StartStream();  
}  
  
objStreaming.EnableDigitalInput () ;  
//  
objStreaming.DisabledDigitalInput () ;
```

See Also

DisableMotionDetection(), OnMDEventStart()

DigitalOutput

Description

Trigger assigned digital out event

Syntax

```
DigitalOutput( LONG nDO1, LONG nDO2, LONG nDO3, LONG nDO4 )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nDO1</i>	LONG	Set to 0 will disable the Digital Output 1, Set to 1 will enable Digital Output 1 signal.
<i>nDO2</i>	LONG	Set to 0 will disable the Digital Output 2, Set to 1 will enable Digital Output 2 signal.
<i>nDO3</i>	LONG	(Reserved) Set to 0 will disable the Digital Output 3, Set to 1 will enable Digital Output 3 signal.
<i>nDO4</i>	LONG	(Reserved) Set to 0 will disable the Digital Output 4, Set to 1 will enable Digital Output 4 signal.

Returns

NONE

Remarks

NONE

Example

```
objDIO.DigitalOutput(1,0,0,0)
```

See Also

Other Methods

<i>Name</i>	<i>Description</i>
-------------	--------------------

4

Control Events

This section lists media events. These values maybe triggered or customized.

UI Events

<i>Name</i>	<i>Description</i>
OnConnect()	Event generated when the control connect success
OnConnectionRecovery()	Event generated when the control reconnect successfully
OnDisconnect()	Event generated when the control disconnect
OnMouseKeyDoubleClick()	Event generated when the user double clicks on the media area in the media window
OnMouseKeyDown()	Event generated when the mouse down on the media area in the media window
OnMouseKeyUp()	Event generated when the mouse up on the media area in the media window
OnMouseMoveNotify()	Event generated when the mouse left button down and move on the media area in the media window
OnNetworkLoss()	Event generated when the connection break.
OnPlay()	Event generated when the control start preview
OnPlaybackComplete()	Event generated when the control playback file complete
OnSaveImage()	Event generated when save a image
OnStartRecord()	Event generated when recording starts
OnStop()	Event generated when the control stop preview
OnStopRecord()	Event generated when recording stops
OnTimeCode()	Event generated when time code was be updated
OnVideoLoss()	Event generated when video loss occurs
OnVideoRecovery()	Event generated when video recovery occurs after video loss

OnConnect()

Description

Event generated when the control connect success.

Syntax

onConnect(LONG *nID*, LONG *ConnectSuccessful*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>nID</i>	Long	Return camera id						
<i>ConnectSuccessful</i>	Long	The connect return code						
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Connect failed</td></tr><tr><td>1</td><td>Connect successful</td></tr></table>			Value	Description	0	Connect failed	1	Connect successful
Value	Description							
0	Connect failed							
1	Connect successful							

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnConnect(nID,successful)" LANGUAGE="Jscript">
    Alert( "Connect Success" );
</SCRIPT>
'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnConnect()
        MsgBox "Connect Success"
    End Function
</SCRIPT>
```

See Also

OnConnectionRecovery()

Description

Event generated when the control connection is reconnected successfully in first time.

Syntax

OnConnectionRecovery(**LONG** *nID*, **BSTR** *szMediaSourceName*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>szMediaSourceName</i>	BSTR	Return media source name

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT FOR=Media EVENT="OnConnectionRecovery()" LANGUAGE="Jscript">
    Alert( "Connection Recovery" );
</SCRIPT>

'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnConnectionRecovery()
        MsgBox "Connection Recovery"
    End Function
</SCRIPT>
```

See Also

OnDisconnect()

Description

Event generated when the control disconnect .

Syntax

OnDisconnect(LONG *m_nID*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>m_nID</i>	LONG	Return camera id

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnDisconnect()" LANGUAGE="Jscript">
    Alert( "Disconnect !!" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnDisconnect()
        MsgBox "Disconnect"
    End Function
</SCRIPT>
```

See Also

OnMouseKeyDoubleClick()

Description

Event generated when the user double clicks on the media area in the media window.

Syntax

OnMouseKeyDoubleClick(LONG *nID*, LONG *nBtn*, LONG *nX*, LONG *nY*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>nBtn</i>	LONG	Determine which mouse button is clicked. 1 is mouse double click. 2 is scroll double click. 3 is right double click.
<i>nX</i>	LONG	X coordinate of the click window, this value will be passed to the hosting application
<i>nY</i>	LONG	Y coordinate of the click window, this value will be passed to the hosting application

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnClickInContent(x,y)" LANGUAGE="Jscript">
    Alert( "Click in content: X(" + x + ") y(" + y + ")" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnClickInContent(x,y)
        MsgBox "Click in content: X(" + x + ") y(" + y + ")"
    End Function
</SCRIPT>
```

See Also

[OnMouseDown\(\)](#), [OnMouseKeyUp\(\)](#), [OnMouseMoveNotify\(\)](#)

OnMouseKeyDown()

Description

Event generated when the mouse down on the media area in the media window.

Syntax

OnMouseKeyDown(LONG *nID*, LONG *nBtn*, LONG *nX*, LONG *nY*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>nBtn</i>	LONG	Determine which mouse button is clicked.
<i>nX</i>	LONG	X coordinate of the click window, this value will be passed to the hosting application
<i>nY</i>	LONG	Y coordinate of the click window, this value will be passed to the hosting application

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnmouseKeyDown(ID, BTN, x,y)" LANGUAGE="Jscript">
    Alert( "Click in content: ID(" + ID + ")BTN(" + BTN + ")X(" + x + ") y(" +
y + ")" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_ OnmouseKeyDown (ID, BTN, x,y)
        MsgBox "Click in content: ID(" + ID + ")BTN(" + BTN + ")X(" + x + ")
y(" + y + ")
    End Function
</SCRIPT>
```

See Also

[OnMouseKeyDoubleClick\(\)](#), [OnMouseKeyUp\(\)](#), [OnMouseMoveNotify\(\)](#)

OnMouseKeyUp()

Description

Event generated when the mouse up on the media area in the media window.

Syntax

OnMouseKeyUp(LONG *nID*, LONG *nBtn*, LONG *nX*, LONG *nY*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return Camera ID
<i>nBtn</i>	LONG	Determine which mouse button is clicked.
<i>nX</i>	LONG	X coordinate of the click window, this value will be passed to the hosting application
<i>nY</i>	LONG	Y coordinate of the click window, this value will be passed to the hosting application

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT FOR=Media EVENT="OnmouseKeyUp(ID, BTN, x, y)" LANGUAGE="Jscript">
    Alert( "Click in content:ID(" + ID + ")BTN(" + BTN + ")X(" + x + ") y(" +
y + ")") );
</SCRIPT>
'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_ OnmouseKeyUp (ID, BTN, x,y)
        MsgBox "Click in content: ID(" + ID + ")BTN(" + BTN + ")X(" + x + ")
y(" + y + ")
    End Function
</SCRIPT>
```

See Also

[OnMouseKeyDoubleClick\(\)](#), [OnMouseKeyDown\(\)](#), [OnMouseMoveNotify\(\)](#)

OnMouseMoveNotify()

Description

Event generated when the mouse left button down and move on the media area in the media window.

Syntax

OnMouseMoveNotify(LONG *nID*, LONG *nX*, LONG *nY*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>nX</i>	LONG	X coordinate of the click window, this value will be passed to the hosting application
<i>nY</i>	LONG	Y coordinate of the click window, this value will be passed to the hosting application

Returns

NONE

Remarks

To use this event, call objStreaming.EnableMouseMoveEvent(1) first.

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnMouseMoveNotify(id, x,y)" LANGUAGE="Jscript">
    Alert( "Click in content: ID(" + id + ")X(" + x + ") y(" + y + ")" );
</SCRIPT>
```

See Also

**OnMouseKeyDoubleClick(), OnMouseKeyDown(), OnMouseKeyUp(),
EnableMouseMoveEvent()**

OnNetworkLoss()

Description

Event generated when the connection break.

Syntax

onNetworkLoss(**LONG** *nID*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT FOR=Media EVENT="OnNetworkLoss(nID)" LANGUAGE="Jscript">
    Alert( nID + " network loss" );
</SCRIPT>

'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnNetworkLoss (nID)
        MsgBox nID + " network loss"
    End Function
</SCRIPT>
```

See Also

OnPlay()

Description

Event generated when the control start preview.

Syntax

onPlay(**LONG** *nID*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnConnect()" LANGUAGE="Jscript">
    Alert( "Connect Success" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnConnect()
        MsgBox "Connect Success"
    End Function
</SCRIPT>
```

See Also

[Play\(\)](#)

OnPlaybackComplete()

Description

Event generated when the control playback file complete.

Syntax

onPlaybackComplete(LONG *nID*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT FOR=Media EVENT="OnPlaybackComplete()" LANGUAGE="Jscript">
    Alert( "PlaybackComplete" );
</SCRIPT>

'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnPlaybackComplete()
        MsgBox "PlaybackComplete"
    End Function
</SCRIPT>
```

See Also

Play()

OnSaveImage()

Description

Event generated when save an image.

Syntax

onSaveImage(**LONG** *nID*, **LONG** *Successful*, **BSTR** *FileName*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>Successful</i>	LONG	Return if save image successful
<i>FileName</i>	BSTR	Return image file name

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnSaveImage()" LANGUAGE="Jscript">
    Alert( "Save image!!" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_ OnSaveImage ()
        MsgBox "Save image!!"
    End Function
</SCRIPT>
```

See Also

OnStartRecord()

Description

Event generated when recording starts.

Syntax

OnStartRecord(LONG *nID*, LONG *nSuccessful*, BSTR *szFileName*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>nSuccessful</i>	LONG	Return if record successful
<i>szFileName</i>	BSTR	Return record file name

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnStartRecord()" LANGUAGE="Jscript">
    Alert( "Start to record!!" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_ OnStartRecord()
        MsgBox "Start to record!!"
    End Function
</SCRIPT>
```

See Also

[OnStopRecord\(\)](#)

OnStop()

Description

Event generated when the control stop preview.

Syntax

onStop(LONG *nID*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnConnect()" LANGUAGE="Jscript">
    Alert( "Connect Success" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnConnect()
        MsgBox "Connect Success"
    End Function
</SCRIPT>
```

See Also

OnStopRecord()

Description

Event generated when recording stops.

Syntax

OnStopRecord(LONG *nID*, BSTR *szFileName*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>szFileName</i>	BSTR	Return record file name

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT FOR=Media EVENT="OnStopRecord()" LANGUAGE="Jscript">
    Alert( "Record Stop!!" );
</SCRIPT>

'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnStopRecord()
        MsgBox "Record Stop!!"
    End Function
</SCRIPT>
```

See Also

[startRecord\(\)](#), [StopRecord\(\)](#)

OnTimeCode()

Description

Event generated when after start media stream and time code updated.

Syntax

OnTimeCode(**LONG** *nID*, **BSTR** *szTimeCode*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>szTimeCode</i>	BSTR	Return camera time

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT=" OnTimeCode(nID, szTimeCode)" LANGUAGE="Jscript">
    Alert( "Record Stop!!" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_ OnTimeCode ()
        MsgBox "Record Stop!!"
    End Function
</SCRIPT>
```

See Also

[StartRecord\(\)](#), [StopRecord\(\)](#)

OnVideoLoss()

Description

Event generated when video loss occurs .

Syntax

onVideLoss(LONG *nID*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT FOR=Media EVENT="OnVideoLoss()" LANGUAGE="Jscript">
    Alert( "Video Loss!!" );
</SCRIPT>

'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnVideoLoss()
        MsgBox "Video Loss!!"
    End Function
</SCRIPT>
```

See Also

[onVideoRecovery\(\)](#)

OnVideoRecovery()

Description

Event generated when video recovery occurs after video loss.

Syntax

onVideRecovery(LONG *nID*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnVideoRecovery()" LANGUAGE="Jscript">
    Alert( "Video Recovery!!" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnVideoRecovery))
        MsgBox "Video Recovery!!"
    End Function
</SCRIPT>
```

See Also

[onVideoLoss\(\)](#)

OnKeyDown()

Description

Event generated when the keyboard down.

Syntax

onKeyDown(LONG *nID*, LONG *key*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>key</i>	LONG	Determine which key is press down.

Returns

NONE

Remarks

NONE

OnKeyUp()

Description

Event generated when the keyboard up.

Syntax

onKeyUp(LONG *nID*, LONG *key*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>key</i>	LONG	Determine which key is press up.

Returns

NONE

Remarks

NONE

Motion Detection Events

<i>Name</i>	<i>Description</i>
OnMDEventStart()	Event triggered when a motion detection event occurs
OnMDEventEnd()	Event triggered when a motion detection event End
OnMDSetting()	Event generated when motion detection setting is enabled
OnMDSettingChange()	Event generated when motion detection setting be changed

OnMDEventStart()

Description

Event generated when motion detection event occurs.

Syntax

OnMDEventStart(LONG *nID*, LONG *nMD*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>nMD</i>	LONG	The sum of regional motion detection numbers triggered. MD1 = 1, MD2 = 2, MD3 = 4. If MD1 and MD3 triggered, the number is 5.

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnMDEventStart(SHORT nMD)" LANGUAGE="JScript">
    Alert( "MD Event: Range " + nMD + " detected !" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnMDEventStart(SHORT nMD)
        MsgBox "MD Event: Range " + nMD + " detected !!"
    End Function
</SCRIPT>
```

See Also

[EnableMotionDetection\(\)](#), [OnMDEventEnd\(\)](#), [MotionDetectionAlertDuration](#),

OnMDEventEnd()

Description

Event generated when motion detection event end.

Syntax

OnMDEventEnd(LONG *nID*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT FOR=Media EVENT="OnMDEventEnd()" LANGUAGE="JScript">
    Alert( "MD Event End !" );
</SCRIPT>

'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnMDEventEnd
        MsgBox "MD Event End !!"
    End Function
</SCRIPT>
```

See Also

EnableMotionDetection(), OnMDEventStart(), MotionDetectionAlertDuration,

OnMDSetting()

Description

Event generated when motion detection setting is enabled.

Syntax

```
OnMDSetting ( LONG nID,  
              LONG nIndex,  
              LONG nX,  
              LONG nY,  
              LONG nW,  
              LONG nH,  
              LONG nSens,  
              LONG enable )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>nIndex</i>	LONG	Return MD index
<i>nX</i>	LONG	Return MD start position in x-axis
<i>nY</i>	LONG	Return MD start position in y-axis
<i>nW</i>	LONG	Return MD width
<i>nH</i>	LONG	Return MD height
<i>nSens</i>	LONG	Return MD sensitive
<i>enable</i>	LONG	Return MD enable status

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT FOR=Media EVENT=" OnMDSetting()" LANGUAGE="Jscript">
    Alert( "Setting the motion!!" );
</SCRIPT>

'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_ OnMDSetting()
        MsgBox "Setting the motion!!"
    End Function
</SCRIPT>
```

See Also

OnMDSettingChange()

Description

Event generated when motion detection setting be changed.

Syntax

OnMDSettingChange(LONG *nID*, LONG *nIndex*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>nIndex</i>	LONG	Return MD index

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT=" OnMDSettingChange()" LANGUAGE="Jscript">
    Alert( "Changing the motion setting!!" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_ OnMDSettingChange()
        MsgBox "Changing the motion setting!!"
    End Function
</SCRIPT>
```

See Also

DIO Events

<i>Name</i>	<i>Description</i>
OnDIEvent()	Event generated when the digital in port is triggered.
OnDIEvent()	Event generated when the digital in port is ended.
OnRS232Event()	Event generated when data comes from RS-232/RS-422/RS-485 port

OnDIEvent()

Description

Event generated when the digital in port is triggered.

Syntax

OnDIEvent(LONG *nID*, LONG *nDI*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>nDI</i>	LONG	Indicates DI port number that is triggered

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnDIEvent(SHORT nDI)" LANGUAGE="JScript">
    Alert( "DI Event: " + nDI + " fired !" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnDIEvent(SHORT nDI)
        MsgBox "DI Event: " + nDI + " fired !!"
    End Function
</SCRIPT>
```

See Also

OnDIEventEnd()

Description

Event generated when the digital in port is ended.

Syntax

OnDIEvent(LONG *nID*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT FOR=Media EVENT="OnDIEvent(SHORT nDI)" LANGUAGE="JScript">
    Alert( "DI Event: " + nDI + " fired !" );
</SCRIPT>

'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnDIEvent(SHORT nDI)
        MsgBox "DI Event: " + nDI + " fired !!"
    End Function
</SCRIPT>
```

See Also

OnRS232Event()

Description

Event generated when data comes from RS-232/RS-422/RS-485 port.

Syntax

OnRS232Event(LONG *nID*, BSTR *szData*, LONG *nDataLen*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera ID
<i>szData</i>	BSTR	The data received from the RS-232/RS-422/RS-485 port
<i>nDataLen</i>	LONG	The length of the data received

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnRS232Event(RS232_Data,iLength)" LANGUAGE="JScript">
    Alert( "RS232 Data: " + RS232_Data + " , len = " + iLength + "!" );
</SCRIPT>

'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnRS232Event(nID, RS232_Data, iLength)
        MsgBox "RS232 Data: " + RS232_Data + " , len = " + iLength + "!"
    End Function
</SCRIPT>
```

See Also

5

New Method

This section lists new implement method.

New Method

<i>Name</i>	<i>Description</i>
SetTextOut()	Set text show on control display screen
MirrorImage()	Reverse image from left to right
FliplImage()	Reverse image for top to bottom
EnablePrivacyMask()	Enable privacy mask function and set the mask color
SetPrivacyMask()	Set privacy mask region
EnableAbsPosition()	Enable absolute PTZ position function
DisableAbsPosition()	Disable absolute PTZ position function
AddMultiplePlaybackFile()	Add file to multiple playback file list.
RemoveAllMultiplePlaybackFile()	Remove all file in the multiple playback file list.
PlayAudioFileToDevice()	Play audio file to device.
GetNumberOfMonitor()	Get number of monitors in os
ShowRealFPS()	Show real fps number on screen
SetRealFpsFontColor()	Set font and color for show real fps
SetTextOutEx()	Set text show on control display screen
ExecuteURLCommand()	Execute URL command
SendURLCmd()	Send URL command
SendPTZCommand()	Send PTZ protocol command to video server or speed dome
RequestsToCheckDeviceChipType()	Request to check device's chip type
EnableFullscreenEx()	Enable the Fullscreen
TxRS232Data()	Send RS232 command
AsyncDisconnect()	Disconnect the media source asynchronously
AsyncSendURLCmd()	Send URL command asynchronously
EnableOnNewImageEvent()	Enable OnMyNewImage() event
SetGlobalMotionEnable()	Set global motion parameter enable or disable
GetGlobalMotionEnable()	Get global motion parameter is enable or disable
SnapshotWithTitleBar()	SnapShot with title bar
GetFullscreenStatus()	Get fullscreen status

CloseFullScreenWindow()	Close the fullscreen mode
GetMouseDigitalPTZStatus()	Get mouse digital PTZ status
GetPlayRate()	Get playback speed
GetPlayDirection()	Get the direction of playback forward or backward
SetUtcCurrentTime()	Sets current time in playback mode use UTC time
SetTextOutToRegion()	Set text show on control display screen
StartRecordEx()	Start play current media stream or media file
StopRecordEx()	Stops recording on the current recording operation
SnapShotEx()	Saves current SnapShot image of current video frame by identity filename
GetCurrentTimeByInteger()	Gets current time in playback mode
SetCurrentTimeByInteger()	Sets current time in playback mode
ConvertUTCtoString()	Convert UTC time to string
EnablePIRMotionDetection()	Enable PIR's motion detection
DisablePIRMotionDetection()	Disable PIR's motion detection
GetPIRConfig()	Get PIR settings
SetPIRConfig()	Set PIR settings
SetMotionTime()	Set the motion timer to the specific motion region
EnableMotionSetting()	Enable to setting motion detection settings
SetMotionSettingEx()	Set motion detection settings
GetMotionInfoFromDevice()	Get motion detection information from device
StartStreamEx()	Starts Decoding mpeg4 stream but don't show in the window
SetDigitalOutput()	Set digital out event enable
DevicePrivacyMask()	Enable/Disable device privacy mask
SnapShotEx4CMS()	Saves current SnapShot image of current video frame by identity filename
GetCurrentImage()	Get current image
GetCurrentImageEx()	Get current image
EnableDecoder()	Set decoder enable or disable
EnableMouseMoveEvent()	Set mouse move event enable or disable
SendRS232Command()	Send RS232 command
RefreshControl()	Refresh control
EnableTitleEvent()	Set title event enable
DisableTitleEvent()	Set title event disable
ExecuteURLCommandEx()	Execute URL command

SetMotionThreshold()	Set motion threshold
DigitalPause()	Digital pause
DigitalPlay()	Digital play
IsLeftButtonUp()	Detect mouse left button up
EnableCurrentImageBuffer()	Enable current image buffer
DisableCurrentImageBuffer()	Disable current image buffer

SetTextOut()

Description

Set text show on control display screen

Syntax

```
LONG SetTextOut( LONG nX,  
                 LONG nY,  
                 BSTR szText,  
                 BSTR szFontName,  
                 SHORT nBold,  
                 LONG nFontWidth,  
                 LONG nnFontHeight,  
                 LONG nFontR,  
                 LONG nFontG,  
                 LONG nFontB )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nX</i>	Long	X coordinate of text
<i>nY</i>	Long	Y coordinate of text
<i>szText</i>	BSTR	String that want to show on screen
<i>szFontName</i>	BSTR	Font name of showed text
<i>nBold</i>	SHORT	Font weight setting.
<i>nFontWidth</i>	LONG	Width of text
<i>nnFontHeight</i>	LONG	Height of text
<i>nFontR</i>	LONG	Red in text color setting
<i>nFontG</i>	LONG	Green in text color setting
<i>nFontB</i>	LONG	Blue in text color setting

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....  
objStreaming.Play()  
.....  
objStreaming.SetTextOut(5,5,"Text out String","Arial",0,10,20,255,255,255) ;  
.....
```

See Also

NONE

MirrorImage()

Description

Reverse image from left to right.

Syntax

`MirrorImage(LONG nEnable)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nEnable</i>	LONG	Set 0 for disable 1 for enable function

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.Play()  
.....  
objStreaming.MirrorImage(0) ;  
.....
```

See Also

NONE

FlipImage()

Description

Reverse image from up to down.

Syntax

```
FlipImage( LONG nEnable )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nEnable</i>	LONG	Set 0 for disable, 1 for enable function

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.Play()  
.....  
objStreaming.FlipImage(1) ;  
.....
```

See Also

NONE

EnablePrivacyMask()

Description

Enable privacy mask function and set mask color

Syntax

EnablePrivacyMask(LONG *nEnable*, LONG *R*, LONG *G*, LONG *B*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nEnable</i>	Long	Set 0 for disable, 1 for enable function
<i>R</i>	Long	Red in RGB color
<i>G</i>	Long	Green in RGB color
<i>B</i>	Long	Blue in RGB color

Returns

NONE

Remarks

NONE

Example

```
.....
objStreaming.Play()
.....
objStreaming.EnablePrivacyMask(1,255,255,255) ;
.....
```

See Also

NONE

SetPrivacyMask()

Description

Set privacy mask region

Syntax

SetPrivacyMask(LONG *nIndex*, LONG *nXStart*, LONG *nYStart*, LONG *nXEnd*, LONG *nYEnd*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nIndex</i>	LONG	Privacy mask index(1~3)
<i>nXStart</i>	LONG	X coordinate of left-top corner
<i>nYStart</i>	LONG	Y coordinate of left-top corner
<i>nXEnd</i>	LONG	X coordinate of right-bottom corner
<i>nYEnd</i>	LONG	Y coordinate of right-bottom corner

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.Play()  
.....  
objStreaming.SetPrivacyMask(3,50,50,150,150) ;  
.....
```

See Also

NONE

EnableAbsPosition()

Description

Enable PTZ absolute position function

Syntax

LONG EnableAbsPosition()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....  
objStreaming.EnablePTZ() ;  
.....  
objStreaming.EnableAbsPosition() ;  
.....
```

See Also

NONE

DisableAbsPosition()

Description

Disable PTZ absolute position function

Syntax

`DisableAbsPosition()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.DisableAbsPosition() ;  
.....
```

See Also

NONE

AddMultiplePlaybackFile()

Description

Add file to multiple playback file list.

Syntax

LONG AddMultiplePlaybackFile(BSTR *szFileName*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szFileName</i>	BSTR	Full path of playback file.(It must has idx file in the same folder)

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....
objStreaming.MediaType = 6
.....
objStreaming.AddMultiplePlaybackFile("C:\RecTest1.RAW")
objStreaming.AddMultiplePlaybackFile("C:\RecTest2.RAW")
.....
```

See Also

NONE

RemoveAllMultiplePlaybackFile()

Description

Remove all file in the multiple playback file list.

Syntax

```
RemoveAllMultiplePlaybackFile()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
objStreaming.MediaType = 6  
.....  
objStreaming.RemoveAllMultiplePlaybackFile()  
.....
```

See Also

NONE

PlayAudioFileToDevice()

Description

Play the audio file to device.

Syntax

LONG PlayAudioFileToDevice(**BSTR** *szFileName*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szFileName</i>	BSTR	Full path of audio file.(It must be 8k 16bit PCM format)

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....
objStreaming.MediaType = 6
.....
objStreaming.PlayAudioFileToDevice("C:\Audio.wav")
.....
```

See Also

StopPlayingAudioFile()

GetNumberOfMonitors()

Description

Get number of monitors in os.

Syntax

LONG GetNumberOfMonitors()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Return number of monitors.

Remarks

NONE

ShowRealFPS()

Description

Show real fps number on screen.

Syntax

```
ShowRealFPS( SHORT bShow )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>bShow</i>	SHORT	1: visible, 2: hidden.

Returns

NONE

Remarks

NONE

SetRealFpsFontColor()

Description

Set font and color for show real fps.

Syntax

```
SetRealFpsFontColor( LONG bBold,  
                     LONG bItalic,  
                     LONG bUnderLine,  
                     BSTR pFontName,  
                     LONG nFontSize,  
                     LONG nTextColorR,  
                     LONG nTextColorG,  
                     LONG nTextColorB,  
                     LONG nBKMode,  
                     LONG nBkR,  
                     LONG nBkG,  
                     LONG nBkB )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>bBold</i>	LONG	Is bold.
<i>bItalic</i>	LONG	Is italic.
<i>bUnderLine</i>	LONG	Is underline.
<i>pFontName</i>	BSTR	Font typeface name.
<i>nFontSize</i>	LONG	Font size.
<i>nTextColorR</i>	LONG	Text color R.
<i>nTextColorG</i>	LONG	Text color G.
<i>nTextColorB</i>	LONG	Text color B.
<i>nBKMode</i>	LONG	Specifies the background mode.
<i>nBkR</i>	LONG	Background color R.
<i>nBkG</i>	LONG	Background color G.
<i>nBkB</i>	LONG	Background color B.

Returns

NONE

Remarks

NONE

SetTextOutEx()

Description

Set font and color for show real fps.

Syntax

```
SetTextOutEx( LONG index,  
              LONG nX,  
              LONG nY,  
              BSTR Text,  
              LONG bBold,  
              LONG bItalic,  
              LONG bUnderLine,  
              BSTR pFontName,  
              LONG nFontSize,  
              LONG nTextColorR,  
              LONG nTextColorG,  
              LONG nTextColorB,  
              LONG nBKMode,  
              LONG nBkR,  
              LONG nBkG,  
              LONG nBkB )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>index</i>	LONG	Specify a index to store data, maximum is 10.
<i>nX</i>	LONG	x-coordinate of pixel.
<i>nY</i>	LONG	y-coordinate of pixel.
<i>Text</i>	BSTR	String, maximum is 1024
<i>bBold</i>	LONG	Is bold.
<i>bItalic</i>	LONG	Is italic.
<i>bUnderLine</i>	LONG	Is underline.
<i>pFontName</i>	BSTR	Font typeface name.
<i>nFontSize</i>	LONG	Font size.
<i>nTextColorR</i>	LONG	Text color R.
<i>nTextColorG</i>	LONG	Text color G.
<i>nTextColorB</i>	LONG	Text color B.
<i>nBKMode</i>	LONG	Specifies the background mode.

<i>nBkR</i>	LONG	Background color R.
<i>nBkG</i>	LONG	Background color G.
<i>nBkB</i>	LONG	Background color B.

Returns

NONE

Remarks

NONE

ExecuteURLCommand ()

Description

Execute URL command.

Syntax

LONG ExecuteURLCommand(**BSTR** *szCommand*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szCommand</i>	BSTR	URL command.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Return zero.

Remarks

NONE

Example

```
.....  
.....  
objStreaming.ExecuteURLCommand("http://192.168.1.2");  
.....
```

See Also

SendURLCmd ()

Description

Send URL command.

Syntax

BSTR SendURLCmd(**BSTR** *szCommand*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szcommand</i>	BSTR	URL command.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Return setting result.

Remarks

NONE

Example

```
.....  
.....  
objStreaming.SendURLCmd ("http://192.168.1.2");  
.....
```

See Also

SendPTZCommand ()

Description

Send PTZ protocol command to video server or speed dome.

Syntax

```
SendPTZCommand( BSTR szHEXStringCommand )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szHEXStringCommand</i>	BSTR	The PTZ command that is defined by protocol file.

Returns

NONE

Remarks

NONE

RequestsToCheckDeviceChipType()

Description

Request to check device's chip type.

Syntax

`RequestsToCheckDeviceChipType(LONG nCheckIt)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nCheckIt</i>	LONG	0: no check 1: check

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.RequestsToCheckDeviceChipType (1);  
.....
```

See Also

EnableFullScreenEx()

Description

Enable the Fullscreen and set to second monitor or not in multi-monitor mode.

Syntax

`EnableFullScreenEx(LONG nUseSecondMonitor)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nUseSecondMonitor</i>	LONG	0: Disable 1: Enable.

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.EnableFullScreenEx (1);  
.....
```

See Also

TxRS232Data()

Description

Send RS232 command.

Syntax

```
LONG TxRS232Data( OLE_HANDLE pData, LONG numBytes )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>pData</i>	OLE_HANDLE	The handle of video.
<i>numBytes</i>	LONG	Command length.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>								
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Success</td></tr><tr><td>-1</td><td>Media type must be : PREVIEW, PREVIEW_RTSP, PREVIEW_RTSP_WOC, MULTICAST, PREVIEW_AHTTP, PREVIEW_AHTTP_PREVIEW, PREVIEW_AHTTP_WITHOUT_CONTROL, PREVIEW_AHTTP_CONTROL_ONLY.</td></tr><tr><td>-2</td><td>Data Len size <=0 or > 4096</td></tr></table>	Value	Description	0	Success	-1	Media type must be : PREVIEW, PREVIEW_RTSP, PREVIEW_RTSP_WOC, MULTICAST, PREVIEW_AHTTP, PREVIEW_AHTTP_PREVIEW, PREVIEW_AHTTP_WITHOUT_CONTROL, PREVIEW_AHTTP_CONTROL_ONLY.	-2	Data Len size <=0 or > 4096
Value	Description									
0	Success									
-1	Media type must be : PREVIEW, PREVIEW_RTSP, PREVIEW_RTSP_WOC, MULTICAST, PREVIEW_AHTTP, PREVIEW_AHTTP_PREVIEW, PREVIEW_AHTTP_WITHOUT_CONTROL, PREVIEW_AHTTP_CONTROL_ONLY.									
-2	Data Len size <=0 or > 4096									

Remarks

NONE

Example

```
.....  
.....  
objStreaming.TxRS232Data (handle,2048);  
.....
```

See Also

AsyncDisconnect()

Description

Disconnect the media source asynchronously.

Syntax

LONG AsyncDisconnect()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....  
.....  
objStreaming.AsyncDisconnect ();  
.....
```

See Also

AsyncSendURLCmd()

Description

Send URL command asynchronously.

Syntax

LONG AsyncSendURLCmd(**LONG** *nIndex*, **BSTR** *szCommand*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nIndex</i>	LONG	0: hidden 1: visible.
<i>szCommand</i>	BSTR	URL Command.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....  
.....  
objStreaming.AsyncSendURLCmd (1);  
.....
```

See Also

EnableOnNewImageEvent()

Description

Enable OnMyNewImage() event.

Syntax

`LONG EnableOnNewImageEvent(LONG nEnable)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nEnable</i>	LONG	0: disable 1: enable.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns zero.

Remarks

NONE

Example

```
.....  
.....  
objStreaming.EnableOnNewImageEvent (1);  
.....
```

See Also

SetGlobalMotionEnable()

Description

Set global motion parameter enable or disable.

Syntax

```
SetGlobalMotionEnable( ULONG bEnable )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>bEnable</i>	ULONG	0: Disable 1: Enable.

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.SetGlobalMotionEnable (1);  
.....
```

See Also

GetGlobalMotionEnable()

Description

Get global motion parameter is enable or disable.

Syntax

```
ULONG GetGlobalMotionEnable()
```

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	ULONG	Return setting value.

Remarks

NONE

Example

```
.....  
.....  
int return_val = objStreaming.GetGlobalMotionEnable();  
.....
```

See Also

SnapshotWithTitleBar ()

Description

Snapshot with title bar.

Syntax

`SnapshotWithTitleBar(ULONG bEnable)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>bEnable</i>	ULONG	0: Disable 1: Enable.

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.SnapshotWithTitleBar(1);  
.....
```

See Also

GetFullScreenStatus()

Description

Get fullscreen status.

Syntax

LONG GetFullScreenStatus()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	0: Normal mode 1: Fullscreen mode.

Remarks

NONE

Example

```
.....  
.....  
int val = objStreaming.GetFullScreenStatus ();  
.....
```

See Also

CloseFullScreenWindow()

Description

Close the fullscreen mode.

Syntax

```
closeFullScreenWindow()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.closeFullScreenWindow ();  
.....
```

See Also

GetMouseDigitalPTZStatus()

Description

Get mouse digital PTZ status.

Syntax

`LONG GetMouseDigitalPTZStatus()`

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	0: Disable 1: Enable.

Remarks

NONE

Example

```
.....  
.....  
long val = objStreaming.GetMouseDigitalPTZStatus ();  
.....
```

See Also

GetPlayRate()

Description

Get playback speed.

Syntax

LONG GetPlayRate()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Playback speed.

Remarks

NONE

Example

```
.....  
.....  
long val = objStreaming.GetPlayRate ();  
.....
```

See Also

GetPlayDirection()

Description

Get the direction of playback forward or backward.

Syntax

LONG GetPlayDirection()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	0: backward 1: forward.

Remarks

NONE

Example

```
.....  
.....  
long val = objStreaming.GetPlayDirection ();  
.....
```

See Also

SetUtcCurrentTime ()

Description

Sets current time in playback mode use UTC time.

Syntax

```
SetUtcCurrentTime( LONG val )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>val</i>	LONG	Current UTC time.

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.SetUtcCurrentTime (123456);  
.....
```

See Also

SetTextOutToRegion()

Description

Set text show on control display screen.

Syntax

```
SetTextOutToRegion(  LONG  index,  
                     LONG  nRegion,  
                     BSTR  Text,  
                     LONG  bBold,  
                     LONG  bItalic,  
                     LONG  bUnderLine,  
                     BSTR  pFontName,  
                     LONG  nFontSize,  
                     LONG  nTextColorR,  
                     LONG  nTextColorG,  
                     LONG  nTextColorB,  
                     LONG  nBKMode,  
                     LONG  nBkR,  
                     LONG  nBkG,  
                     LONG  nBkB  )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>index</i>	LONG	Line number is range from 0~15.
<i>nRegion</i>	LONG	Line number is range from 0~15.
<i>Text</i>	BSTR	Title text
<i>bBold</i>	LONG	0:Non Bold 1:Bold.
<i>bItalic</i>	LONG	0:Non Italic 1:Italic.
<i>bunderLine</i>	LONG	0:Non UnderLine 1: UnderLine.
<i>pFontName</i>	BSTR	Set Font style.
<i>nFontSize</i>	LONG	Set Font size.
<i>nTextColorR</i>	LONG	Text color(0~255).
<i>nTextColorG</i>	LONG	Text color(0~255).
<i>nTextColorB</i>	LONG	Text color(0~255).
<i>nBKMode</i>	LONG	Background mode.
<i>nBkR</i>	LONG	Background color(0~255).
<i>nBkG</i>	LONG	Background color(0~255).
<i>nBkB</i>	LONG	Background color(0~255).

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.SetTextOutToRegion (1, 1, "Test", 0, 0,0, "Arial", 20, 255, 255,  
255, 3, 0, 0, 0);  
.....
```

See Also

StartRecordEx()

Description

Start recording current media stream or media file.

Syntax

LONG StartRecordEx(**BSTR** *szFileFolder*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szFileFolder</i>	BSTR	Set recordind file saving folder.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....  
.....  
objStreaming.StartRecordEx ("C://");  
.....
```

See Also

StopRecordEx()

Description

Stops recording on the current recording operation.

Syntax

`StopRecordEx()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming. StopRecordEx ();  
.....
```

See Also

SnapShotEx()

Description

Saves current SnapShot image of current video frame by identity filename.

Syntax

```
BSTR SnapShotEx( LONG IFormat,  
                 BSTR FileFolder,  
                 LONG ReduplicateTitleBarText,  
                 LONG R_TextColor,  
                 LONG G_TextColor,  
                 LONG B_TextColor )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>IFormat</i>	LONG	0:BMP 1:JPG 2:GIF 3:TIFF 4:PNG.
<i>FileFolder</i>	BSTR	Set SnapShot file saving folder.
<i>ReduplicateTitleBarText</i>	LONG	Specifies the word to the picture reduplicate.
<i>R_TextColor</i>	LONG	Specifies the red color to the picture adjust.
<i>G_TextColor</i>	LONG	Specifies the green color to the picture adjust.
<i>B_TextColor</i>	LONG	Specifies the blue color to the picture adjust.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns null.

Remarks

NONE

Example

```
.....  
.....  
objStreaming.SnapShotEx (1, "c:\RecTest\SnapShot" & nFile & ".jpg", 1, r_Color,  
g_color, b_color);  
.....
```

See Also

GetCurrentTimeByInteger()

Description

Gets current time in playback mode.

Syntax

LONG GetCurrentTimeByInteger()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Current time.

Remarks

NONE

Example

```
.....  
.....  
int val = objStreaming.GetCurrentTimeByInteger ();  
.....
```

See Also

SetCurrentTimeByInteger()

Description

Sets current time in playback mode.

Syntax

```
SetCurrentTimeByInteger( LONG nCurrentTime )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nCurrentTime</i>	LONG	Set current time by integer.

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.SetCurrentTimeByInteger (123456);  
.....
```

See Also

ConvertUTCtoString()

Description

Convert UTC time to string.

Syntax

BSTR ConvertUTCtoString(**LONG** *nTimeStamp*, **BSTR** *szFormat*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nTimeStamp</i>	LONG	Time stamp.
<i>szFormat</i>	BSTR	Time Format.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Time string

Remarks

NONE

Example

```
.....  
.....  
objStreaming.ConvertUTCtoString (123456,"UTC");  
.....
```

See Also

EnablePIRMotionDetection()

Description

Enable PIR's motion detection.

Syntax

```
EnablePIRMotionDetection()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.EnablePIRMotionDetection ();  
.....
```

See Also

DisablePIRMotionDetection()

Description

Disable PIR's motion detection.

Syntax

```
DisablePIRMotionDetection()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.DisablePIRMotionDetection ();  
.....
```

See Also

GetPIRConfig()

Description

Get PIR settings.

Syntax

BSTR GetPIRConfig()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Success: "bEnable, dwSensitive, dwTime". Fail: "0,0,0"

Remarks

NONE

Example

```
.....  
.....  
BSTR str = objStreaming.GetPIRConfig ();  
.....
```

See Also

SetPIRConfig()

Description

Set PIR settings.

Syntax

`ULONG SetPIRConfig(ULONG bEnable, ULONG nSensitive, ULONG nTime)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>bEnable</i>	ULONG	0: Disable 1: Enable.
<i>nSensitive</i>	ULONG	current motion sensitive value.
<i>nTime</i>	ULONG	current motion timer value.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_value</i>	ULONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>1</td><td>Success</td></tr></table>	Value	Description	0	Fail	1	Success
Value	Description							
0	Fail							
1	Success							

Remarks

NONE

Example

```
.....  
.....  
objstreaming.SetPIRConfig (1,80,10);  
.....
```

See Also

SetMotionTime()

Description

Set the motion timer to the specific motion region.

Syntax

```
SetMotionTime( LONG nMotionRegionNumber, LONG nTime )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nMotionRegionNumber</i>	LONG	Set the special the motion region number.
<i>nTime</i>	LONG	current motion timer value.

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.SetMotionTime (1,10);  
.....
```

See Also

EnableMotionSetting()

Description

Enable to setting motion detection settings.

Syntax

`ULONG EnableMotionSetting(ULONG bEnable)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>bEnable</i>	ULONG	0: Disable 1: Enable.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	ULONG	Return one.

Remarks

NONE

Example

```
.....  
.....  
objStreaming.EnableMotionSetting (1);  
.....
```

See Also

SetMotionSettingEx()

Description

Set motion detection settings.

Syntax

```
SetMotionSettingEx( LONG nMotionRegionNumber,  
                    LONG nStartX,  
                    LONG nStartY,  
                    LONG nEndX,  
                    LONG nEndY,  
                    LONG nSensitive,  
                    LONG nTime,  
                    LONG bEnable )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nMotionRegionNumber</i>	LONG	Motion detection window number. Valid values are: 1, 2, 3
<i>nStartX</i>	LONG	Motion detection window region starting X value
<i>nStartY</i>	LONG	Motion detection window region starting Y value
<i>nEndX</i>	LONG	Motion detection window region ending X value
<i>nEndY</i>	LONG	Motion detection window region ending Y value
<i>nSensitive</i>	LONG	Motion detection window region sensitivity
<i>nTime</i>	LONG	Motion detection window region timer
<i>bEnable</i>	LONG	Motion detection window region enable

Returns

NONE

Remarks

NONE

Example

```
objStreaming.SetMotionSettingEx (1,0,0,100,100,80,10,1);
```

See Also

GetMotionInfoFromDevice()

Description

Get motion detection information from device.

Syntax

BSTR GetMotionInfoFromDevice()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	MOTION_ENABLED="0x01" MOTION_CONFIG="Region ,number,bEnable, StartX,StartY,EndX,EndY, Sensitive,Timer, Threshold"

Remarks

NONE

Example

```
.....  
.....  
BSTR str = objStreaming.GetMotionInfoFromDevice ();  
.....
```

See Also

StartStreamEx()

Description

Starts Decoding mpeg4 stream but don't show in the window.

Syntax

LONG StartStreamEx()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	0: fail 1: success.

Remarks

NONE

Example

```
.....  
.....  
long val = objStreaming.StartStreamEx ();  
.....
```

See Also

SetDigitalOutput()

Description

Set digital out event enable.

Syntax

`setDigitalOutput(LONG nDOIndex, LONG enable)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nDOIndex</i>	LONG	DI index.
<i>enable</i>	LONG	0:disable 1:enable

Returns

NONE

Remarks

NONE

Example

```
.....  
.....  
objStreaming.SetDigitalOutput (1,1);  
.....
```

See Also

DevicePrivacyMask()

Description

Enable/Disable device privacy mask.

Syntax

```
BSTR DevicePrivacyMask( LONG nEnable )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nEnable</i>	LONG	0: disable 1: enable.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Return null.

Remarks

NONE

Example

```
.....  
.....  
objStreaming.DevicePrivacyMask (1);  
.....
```

See Also

SnapShotEx4CMS()

Description

Saves current SnapShot image of current video frame by identity filename.

Syntax

```
BSTR SnapShotEx4CMS( LONG IFormat,  
                    LPCTSTR FileFolder,  
                    LONG ReduplicateTitleBarText,  
                    LONG R_TextColor,  
                    LONG G_TextColor,  
                    LONG B_TextColor,  
                    LPCTSTR NVRName,  
                    LPCTSTR CamName )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>IFormat</i>	LONG	0:BMP 1:JPG 2:GIF 3:TIF 4:PNG.
<i>FileFolder</i>	LPCTSTR	Set SnapShot file saving folder.
<i>ReduplicateTitleBarText</i>	LONG	Specifies the word to the picture reduplicate.
<i>R_TextColor</i>	LONG	Specifies the red color to the picture adjust.
<i>G_TextColor</i>	LONG	Specifies the green color to the picture adjust.
<i>B_TextColor</i>	LONG	Specifies the blue color to the picture adjust.
<i>NVRName</i>	LPCTSTR	NVR name
<i>CamName</i>	LPCTSTR	Cam name

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Return null.

Remarks

This function is for CMS's SnapShot.

Example

```
.....  
.....  
objStreaming. SnapShotEx4CMS (1, "c:\RecTest\SnapShot" & nFile & ".jpg", 1,  
r_Color, g_color, b_color,"NVR1","Cam1");  
.....
```

See Also

GetCurrentImage()

Description

Get current image.

Syntax

```
LONG GetCurrentImage( BYTE* pBuffer, LONG nBufferLen )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>pBuffer</i>	BYTE*	Save image buffer.
<i>nBufferLen</i>	LONG	Image size.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>else</td><td>The image size.</td></tr></table>	Value	Description	0	Fail	else	The image size.
Value	Description							
0	Fail							
else	The image size.							

Remarks

NONE

Example

See Also

GetCurrentImageEx()

Description

Get current image.

Syntax

```
LONG GetCurrentImageEx( OLE_HANDLE Int32BufferPointer, LONG nBufferLen )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Int32BufferPointer</i>	OLE_HANDLE	Save image buffer.
<i>nBufferLen</i>	LONG	Image size.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Return_Value</i>	LONG	Returns setting result. <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Fail</td></tr><tr><td>else</td><td>The image size.</td></tr></table>	Value	Description	0	Fail	else	The image size.
Value	Description							
0	Fail							
else	The image size.							

Remarks

NONE

Example

See Also

EnableDecoder()

Description

Set decoder enable or disable.

Syntax

```
LONG EnableDecoder( LONG nEnable )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nEnable</i>	LONG	0: Disable, 1: Enable.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns zero.

Remarks

NONE

Example

See Also

EnableMouseMoveEvent()

Description

Set mouse move event enable or disable.

Syntax

```
EnableMouseMoveEvent( LONG nEnable )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nEnable</i>	LONG	0: Disable, 1: Enable.

Returns

NONE

Remarks

NONE

Example

See Also

SendRS232Command()

Description

Send rs232 command.

Syntax

```
SendRS232Command( LPCTSTR szCommand )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szCommand</i>	LPCTSTR	RS232 command.

Returns

NONE

Remarks

NONE

Example

See Also

RefreshControl()

Description

Refresh control.

Syntax

```
refreshControl()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

See Also

EnableTitleEvent()

Description

Set title event enable.

Syntax

```
LONG EnableTitleEvent()
```

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Return zero.

Remarks

NONE

Example

See Also

DisableTitleEvent()

Description

Set title event disable.

Syntax

```
DisableTitleEvent()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

See Also

ExecuteURLCommandEx()

Description

Execute URL command.

Syntax

BSTR ExecuteURLCommandEx(**BSTR** szCommand)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>szCommand</i>	BSTR	URL command.

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Return null.

Remarks

NONE

Example

```
.....  
.....  
objStreaming.ExecuteURLCommandEX("http://192.168.1.2");  
.....
```

See Also

SetMotionThreshold()

Description

Set motion threshold.

Syntax

SetMotionThreshold(LONG nMotionRegionNumber, LONG nThreshold)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
nMotionRegionNumber	LONG	Motion region number.
nThreshold	LONG	Threshold.

Returns

NONE

Remarks

NONE

Example

See Also

DigitalPause()

Description

Digital pause.

Syntax

`DigitalPause()`

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

See Also

DigitalPlay()

Description

Digital play.

Syntax

```
digitalPlay()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

See Also

IsLeftButtonUp()

Description

Detect mouse left button up.

Syntax

LONG IsLeftButtonUp()

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	0: Down, 1: Up.

Remarks

NONE

Example

See Also

EnableCurrentImageBuffer()

Description

Enable current image buffer.

Syntax

```
EnableCurrentImageBuffer()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

See Also

DisableCurrentImageBuffer()

Description

Disable current image buffer.

Syntax

```
DisableCurrentImageBuffer()
```

Parameters

NONE

Returns

NONE

Remarks

NONE

Example

See Also

6

New Property

This section lists new implement property.

New Property

<i>Name</i>	<i>Description</i>
MotionDetectionInterval	[I/O] Sets/Gets motion detection interval
RenderInterface	[I/O] Sets/Gets current render using interface type
PlayFileName	[I/O] Sets/Gets current playback file name
RealFPS	[O] Gets current fps value
RecordLedStatus	[I/O] Sets/Gets recording LED status
RecordStatus	[O] Gets record status
SiteID	[I/O] Sets/Gets site ID
PIRDetectionAlertDuration	[I/O] Sets/Gets PIR detection alert duration
DIDefaultValue	[I/O] Sets/Gets DI default value
Is4VGAMode	[I/O] Sets/Gets the 4VGA mode value
NVRID	[I/O] Sets/Gets NVR ID
NVRName	[I/O] Sets/Gets NVR name

MotionDetectionInterval

Description

[I/O] Sets/Gets motion detection interval

Syntax

MotionDetectionInterval = *Input_value*

Return_value = **MotionDetectionInterval** [*default: 10*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	LONG	Sets motion detection interval

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Gets motion detection interval

Remarks

NONE

Example

```
..  
.. ..  
objStreaming.MotionDetectionInterval = 10;
```

See Also

RenderInterface

Description

[I/O] Sets/Gets current render using interface type

Syntax

RenderInterface = *Input_value*

Return_value = **RenderInterface** [*default: 0*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Input_value</i>	LONG	Sets current render using interface type						
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>DGDI</td></tr><tr><td>1</td><td>DXDRAW</td></tr></table>	Value	Description	0	DGDI	1	DXDRAW
Value	Description							
0	DGDI							
1	DXDRAW							

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns current render using interface type

Remarks

NONE

Example

See Also

PlayFileName

Description

[I/O] Sets/Gets current playback file name

Syntax

```
PlayFileName = "Input_Value"  
"Return_Value" = PlayFileName
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	BSTR	Sets current playback file name

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	BSTR	Gets current playback file name

Remarks

NONE

Example

```
objStreaming.PlayFileName = "TEST";
```

See Also

RealFPS

Description

[o] Gets current fps value

Syntax

Return_Value = **RealFPS**

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	ULONG	Returns current fps value

Remarks

NONE

Example

```
int val = ObjStreaming.RealFPS;
```

See Also

RecordLedStatus

Description

[I/O] Sets/Gets recording LED status

Syntax

RecordLedStatus = *Input_value*

Return_value = **RecordLedStatus** [*default: 0*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Input_value</i>	LONG	Sets recording LED status						
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>Normal</td></tr><tr><td>1</td><td>RED LED</td></tr></table>			Value	Description	0	Normal	1	RED LED
Value	Description							
0	Normal							
1	RED LED							

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Gets recording LED status

Remarks

NONE

Example

See Also

RecordStatus

Description

[o] Gets recording status

Syntax

Return_Value = **RecordStatus**

Parameters

NONE

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_Value</i>	LONG	Returns recording status. 0: no record 1: recording

Remarks

NONE

Example

See Also

SiteID

Description

[I/O] Sets/Gets site ID

Syntax

SiteID = *"Input_value"*
"Return_value" = **SiteID**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	BSTR	Sets site ID

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BSTR	Gets site ID

Remarks

NONE

Example

See Also

PIRDetectionAlertDuration

Description

[I/O] Sets/Gets PIR detection alert duration

Syntax

PIRDetectionAlertDuration = *Input_value*

Return_value = **PIRDetectionAlertDuration** [*default: 10*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	LONG	Sets PIR detection alert duration

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Returns PIR detection alert duration

Remarks

NONE

Example

```
objStreaming.PIRDetectionAlertDuration = 10;
```

See Also

DIDefaultValue

Description

[I/O] Sets/Gets DI default value

Syntax

DIDefaultValue = *Input_Value*

Return_value = **DIDefaultValue** [*default: 0x03*]

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>Input_value</i>	BYTE	Sets the DI default value						
		<table><tr><th>Value</th><th>Description</th></tr><tr><td>0x00</td><td>DI_DEFAULT_IS_LOW</td></tr><tr><td>0x03</td><td>DI_DEFAULT_IS_HIGH</td></tr></table>	Value	Description	0x00	DI_DEFAULT_IS_LOW	0x03	DI_DEFAULT_IS_HIGH
Value	Description							
0x00	DI_DEFAULT_IS_LOW							
0x03	DI_DEFAULT_IS_HIGH							

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BYTE	Returns the DI default value

Remarks

NONE

Example

See Also

Is4VGAMode

Description

[I/O] Sets/Gets the 4VGA mode value

Syntax

```
Is4VGAMode = Input_Value  
Return_Value = Is4VGAMode [ default: FALSE ]
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	BOOL	Sets the 4VGA mode value

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	BOOL	Returns current setting value

Remarks

NONE

Example

```
objStreaming.Is4VGAMode = TRUE;
```

See Also

NVRID

Description

[I/O] Sets/Gets NVR ID

Syntax

NVRID = *Input_value*

Return_value = **NVRID**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_value</i>	LONG	Sets NVR ID

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	LONG	Gets NVR ID

Remarks

NONE

Example

See Also

NVRName

Description

[I/O] Sets/Gets NVR name

Syntax

NVRName = *Input_Value*
Return_Value = **NVRName**

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Input_Value</i>	BSTR	Sets NVR Name

Returns

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>Return_value</i>	Bstr	Gets NVR name

Remarks

NONE

Example

See Also

7

New Events

This section lists new implement events.

New Events

<i>Name</i>	<i>Description</i>
OnPlayAudioFileComplete()	Event generated when the audio file play complete
OnMouseMove()	Event generated when the mouse move on the media area in the media window
OnAsyncURLCommandDone()	Event generated when the URL command send
OnTimeZoneEx()	Event generated when set Time Zone
OnTimeCodeEx()	Event generated when set Time code
OnMDSSettingEx()	Event generated when receive motion setting
OnPIREventStart()	Event generated when PIR event start
OnPIREventEnd()	Event generated when PIR event end
RequestSetIOnly()	Event generated when request decode I only
OnDIEventEx()	Event generated when the DI is triggered
OnNewImage()	Event generated when new image
OnAutoReconnect	Event generated when auto-reconnect

OnPlayAudioFileComplete()

Description

Event generated when the audio file play complete.

Syntax

`onPlayAudioFileComplete(LONG nID)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnPlayAudioFileComplete(nID)" LANGUAGE="Jscript">
    Alert( "Play Complete" );
</SCRIPT>
'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_PlayAudioFileComplete()
        MsgBox "Play Complete"
    End Function
</SCRIPT>
```

See Also

OnMouseMove()

Description

Event generated when the mouse move on the media area in the media window.

Syntax

OnMouseMove(LONG *nID*, LONG *nX*, LONG *nY*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	Long	Return camera id
<i>nX</i>	Long	X position
<i>nY</i>	LONG	Y position

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnMouseMove(nID,nX,nY)" LANGUAGE="Jscript">
    Alert( "Move" );
</SCRIPT>
'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnMouseMove()
        MsgBox "Move"
    End Function
</SCRIPT>
```

See Also

OnAsyncURLCommandDone()

Description

Event generated when the URL command send.

Syntax

OnAsyncURLCommandDone(LONG *nID*, LONG *nIndex*, BSTR *szResult*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id
<i>nIndex</i>	LONG	Command index
<i>szResult</i>	BSTR	Result

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnAsyncURLCommandDone(nID,nIndex,szResult)"
LANGUAGE="Jscript">
    Alert( "Send success" );
</SCRIPT>
'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnAsyncURLComandDone()
        MsgBox "Send Success"
    End Function
</SCRIPT>
```

See Also

OnTimeZoneEx()

Description

Event generated when Time Zone updates.

Syntax

`OnTimeZoneEx(LONG nID, LONG timeval, LONG timeZone, LONG fixTimeZone)`

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id
<i>timeval</i>	LONG	Return camera time
<i>timeZone</i>	LONG	Return camera timezone
<i>fixTimeZone</i>	LONG	Return camera fix timezone

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnTimeZoneEx(nID,timeval,timezone,fixtimezone)"
LANGUAGE="Jscript">
    Alert( "set success" );
</SCRIPT>
'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnTimeZoneEx()
        MsgBox "set Success"
    End Function
</SCRIPT>
```

See Also

OnTimeCodeEx()

Description

Event generated when set Time code.

Syntax

OnTimeCodeEx(LONG *nID*, LONG *nTimeCode*, LONG *nMSec*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id
<i>nTimeCode</i>	LONG	Return camera time
<i>nMSec</i>	LONG	Return camera time million second

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT          FOR=Media          EVENT="OnTimecodeEx(nID,nTimeCode,nMSec)"
LANGUAGE="Jscript">
    Alert( "set Success" );
</SCRIPT>
'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnTimeCodeEx()
        MsgBox "set Success"
    End Function
</SCRIPT>
```

See Also

OnMDSettingEx()

Description

Event generated when receive motion setting.

Syntax

```
OnMDSettingEx( LONG nID,  
               LONG nIndex,  
               LONG nX,  
               LONG nY,  
               LONG nW,  
               LONG nH,  
               LONG nSens,  
               LONG enable,  
               LONG nTime )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id
<i>nIndex</i>	LONG	Motion region index
<i>nX</i>	LONG	X position
<i>nY</i>	LONG	Y position
<i>nW</i>	LONG	Motion region width
<i>nH</i>	LONG	Motion region height
<i>nSens</i>	LONG	Motion region sensitivity
<i>enable</i>	LONG	Enable or disable
<i>nTime</i>	LONG	Motion region timer

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT
EVENT="OnMDSettingEx(nID,nIndex,nX,nY,nW,nH,nSens,enable,nTime)"
LANGUAGE="Jscript">
    Alert( "Motion Success" );
</SCRIPT>
'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnMDSettingEx()
        MsgBox "Motion settings Success"
    End Function
</SCRIPT>
```

FOR=Media

See Also

OnPIREventStart()

Description

Event generated when PIR event start.

Syntax

```
OnPIREventStart( LONG nID )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT FOR=Media EVENT="OnPIREventStart(nID)" LANGUAGE="Jscript">
    Alert( "PIR Start" );
</SCRIPT>
'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnPIREventStart()
        MsgBox "PIR Start"
    End Function
</SCRIPT>
```

See Also

OnPIREventEnd()

Description

Event generated when PIR event end.

Syntax

OnPIREventEnd(LONG *nID*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnPIREventEnd(nID)" LANGUAGE="Jscript">
    Alert( "PIR End" );
</SCRIPT>
'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnPIREventEnd()
        MsgBox "PIR End"
    End Function
</SCRIPT>
```

See Also

RequestSetIOnly()

Description

Event generated when request decode I only.

Syntax

RequestSetIOnly(**LONG** *nID*, **LONG** *bRequestSetIOnly*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>nID</i>	Long	Return camera id						
<i>bRequestSetIOnly</i>	Long	The set I only <table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>No set I only</td></tr><tr><td>1</td><td>Set I only</td></tr></table>	Value	Description	0	No set I only	1	Set I only
Value	Description							
0	No set I only							
1	Set I only							

Returns

NONE

Remarks

NONE

Example

```
'----- Java Script
<SCRIPT      FOR=Media      EVENT="      RequestSetIOnly(nID,bRequestSetIOnly)"
LANGUAGE="Jscript">
    Alert( "set I only" );
</SCRIPT>
'----- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_RequestsetIOnly()
        MsgBox "set I only"
    End Function
</SCRIPT>
```

See Also

OnDIEventEx()

Description

Event generated when the DI is triggered.

Syntax

OnDIEventEx(LONG *nDIStatus*)

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>						
<i>nDIStatus</i>	LONG	Return DI status						
<table><tr><th>Value</th><th>Description</th></tr><tr><td>0</td><td>close</td></tr><tr><td>1</td><td>open</td></tr></table>			Value	Description	0	close	1	open
Value	Description							
0	close							
1	open							

Returns

NONE

Remarks

NONE

Example

```
'---- Java Script
<SCRIPT FOR=Media EVENT="OnDIEventEx(nDIStatus)" LANGUAGE="Jscript">
    Alert( "DI" );
</SCRIPT>
'---- VB Script
<SCRIPT LANGUAGE="VBScript">
    Function Media_OnDIEventEx()
        MsgBox "DI"
    End Function
</SCRIPT>
```

See Also

OnNewImage()

Description

Event generated when new image.

Syntax

```
OnNewImage( LONG nID )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id

Returns

NONE

Remarks

NONE

OnAutoReconnect()

Description

Event generated when auto-reconnect

Syntax

```
onNewImage( LONG nID )
```

Parameters

<i>Name</i>	<i>Type</i>	<i>Description</i>
<i>nID</i>	LONG	Return camera id

Returns

NONE

Remarks

NONE