

API CGI

1 Global

1.1 ABBREVIATIONS

The following abbreviations are used throughout this document.

| | |
|------------|---|
| CGI | Common Gateway Interface - a standardized method of communication between a client (e.g. a web browser) and a server (e.g. a web server). |
| URL | RFC 1738 describes the syntax and semantics for a compact string representation for a resource available via the Internet. These strings are called "Uniform Resource Locators" (URLs). |
| URI | A Uniform Resource Identifier (URI) is a compact string of characters for identifying an abstract or physical resource. RFC 2396 describes the generic syntax of URI. |

1.2 TRANSACTION

The HTTP API Transaction starts from a request from a client Application, usually a web browser. The request is processed by the web server on the IP Camera. Then send the response back to the client application. The HTTP APP is taken in GET form. If the request is successful, the IP Camera will return a HTTP header contains 200 OK. The HTTP Body will contain actual data or error message if an error occurs.

For describe convenience, we use some short words to instead the long expressions. The follows are several regulations:

1. The italics and bold will be replaced by the value behind the symbol “=”.
2. The URL must follow the standard way of writing a URL.(RFC_3986:Uniform Resource Identifiers (URI) Generic Syntax);that is ,spaces and other reserved characters (“;”, “/”, “?”, “:”, “@”, “=”, “+”, “,” and “\$”) within a <paramName> or a <paramValue> must be replaced with %<ASCII hex>.For example ,the blank must be instead with %20.
3. To describe the range of the parameter, we use some symbols such as “[]”, “{ }” and so on. For example :”[0-100]”denotes a integer not less than 0 and not larger than 100. “{0,1,2,3}”denotes the valid value of a integer among 0,1,2 and 3.
4. In the request and response, we use “[]” to denote an array. The index is usually a integer and start form 0.
5. The parameter value has several types: string, integer, bool and float.Integer is 32 bits.The range of bool is “true” and “false”.

2AUTHENTICATION

The IP Camera supplies two authentication ways: basic authentication and digest authentication. The basic authentication is encrypted with base64 and the digest authentication is encrypted with MD5. When client sent the information without authorized or other conditions make the session can not build, the servers will return HTTP Code to give client some information. The description of HTTP Code is in the below table.

| HTTP Code | HTTP Text | Description |
|------------|------------------------|---|
| 200 | OK | The request has succeeded, but an application error can still occur, which will be returned as an application error code. |
| 204 | No Content | The server has fulfilled the request, but there is no new information to send back. |
| 302 | Moved Temporarily | The server redirects the request to the URI given in the Location header. |
| 400 | Bad Request | The request had bad syntax or was impossible to fulfill. |
| 401 | Unauthorized | The request requires user authentication or the authorization has been refused. |
| 404 | Not Found | The server has not found anything matching the request. |
| 409 | Conflict | The request could not be completed due to a conflict with the current state of the resource. |
| 500 | Internal Error | The server encountered an unexpected condition that prevented it from fulfilling the request. |
| 503 | Service Unavailable | The server is unable to handle the request due to temporary overload. |

Example: Request includes unauthorized.

```
HTTP/1.1 401 Unauthorized
WWW-Authenticate: Basic realm="Device_CGI"
CONNECTION: close
CONTENT-LENGTH: 0
```

1. When basic authentication, the IP camera response:

401 Unauthorized (说明是未认证的)

WWW-Authenticate: Basic realm="XXXXXX"

Then the client encode the username and password with base64, send the following request:

Authorization: Basic VXZVXZ.

2. When digest authentication, the IP camera response:

WWW-Authenticate: Digest realm="ZENO_00408CA5EA04",

nonce="000562fdY631973ef04f77a3ede7c1832ff48720ef95ad",

stale=FALSE,
qop="auth";

The client calculates the digest using username, password, nonce, realm and URI with MD5, then send the following request:

```
Authorization:Digestusername="admin", realm="ZENO_00408CA5EA04", nc=00000001,  
cnonce="0a4f113b", qop="auth" nonce="000562fdY631973ef04f77a3ede7c1832ff48720  
ef95ad", uri="/cgi-bin/system.cgi?action=getMaxExtraStream", response="65002d  
e02df697e946b750590b44f8bf"
```

3 VIDEO ENCODE

3.1. Stream

3.1.1 Get Max Extra Stream Counts

Get the max counts of extra stream.

Method:GET

Syntax:

```
http://<ip>/cgi-bin/system.cgi?action=getMaxExtraStream
```

Response:

```
table.MaxExtraStream=1
```

comment:

```
The range of table.MaxExtraStream is {1,2,3}
```

3.1.2 Get MJPG Stream

Get MJPG stream

Method: GET

Syntax:

```
http://<ip>/cgi-bin/mjpg/video.cgi?channel=<channelNo>&subtype=<typeNo>
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|------------------|-----------------|---|
| ip | | Device ip address |
| channelNo | | Channel number, since this is not used now, the value is written as 0 by default. |
| typeNo | | Flow type, 0: primary stream, 1: secondary stream 1, 2: secondary stream 2 |

Response:

```
OK or ERROR  
If OK
```

Content-Type: multipart/x-mixed-replace; boundary=firstboundary;

--firstboundary

Content-Type: image/jpeg

Content-Length: <mjpg octec stream length>

<mjpg octet stream>

comment:

Get a mjpg stream from a device

3.2 Video Color

3.2.1 Set Parameter

Set the parameter of video color.

Method:GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | ParamValue type | Description |
|--------------------------|-----------------|---|
| <i>head</i> .Brightness | Integer | Brightness, range is [0-100] |
| <i>head</i> .Contrast | Integer | Contrast, range is [0-100] |
| <i>head</i> .Hue | Integer | Hue, range is [0-100] |
| <i>head</i> .Saturation | Integer | Saturation, range is [0-100] |
| <i>head</i> .TimeSection | String | Effective time for this video color param. Format is: <i>mask starttimeendtime</i> Mask range is {0, 1}. Mask 0 – this video param is not effective Mask 1 - this param is effective <i>Starttime/Endtime</i> format like 11:00:00. Example: |

| | | |
|--|--|--|
| | | <p>0 01:00:00-02:00:00, means this param is not effective.</p> <p>1 01:00:00-02:00:00, means this param is effective between 01:00:00 and 02:00:00</p> |
|--|--|--|

comment:

In above table, **head**=VideoColor[*ChannelNo*][*ColorParamNo*]

ChannelNo = video channel index,

ColorParamNo = color Param index,

0 = Color Param 1

1 = Color Param 2

...

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&VideoColor[0][0].Brightness=2

3.2.2 Get Parameter

Get the parameter of video color.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=**VideoColor**

Response:

head.Brightness=50

head.Contrast=50

head.Hue=50

head.Saturation=50

head.TimeSection=1 00:00:00-24:00:00

comment:

In above table, **head**=table.VideoColor[*ChannelNo*][*ColorParamNo*]

ChannelNo = video channel index,

ColorParamNo = color param index.

0 = Color Param 1

1 = Color Param 2

...

3.3 Video In Options

3.3.1 Get Capabilities

Get video input capabilities.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/videoInput.cgi?action=getCaps&channel=<<channelNo>
```

Response:

```
caps.Backlight=true  
caps.ChipID=0  
caps.CoverCount=0  
caps.CoverType=0  
caps.CustomManualExposure=true  
caps.DayNightColor=true  
caps.DownScaling=true  
caps.Exposure=9  
caps.ExternalSyncInput=true  
caps.FlashAdjust=true  
caps.Flip=true  
caps.Gain=true  
caps.GainAuto=true  
caps.HorizontalBinning=1  
caps.InfraRed=false  
caps.Iris=false  
caps.IrisAuto=false  
caps.LadenBitrate=4096  
caps.LimitedAutoExposure=true  
caps.MaxHeight=960  
caps.MaxWidth=1280  
caps.Mirror=false  
caps.NightOptions=false  
caps.ReferenceLevel=false  
caps.Rotate90=false  
caps.SetColor=true  
caps.SignalFormats=BT656
```

```
caps.SyncChipChannels=false
caps.TitleCount=0
caps.UpScaling=false
caps.VerticalBinning=0
caps.WhiteBalance=3
```

With the following parameter and value type:

| Field In Respons | Param Value type | Description |
|----------------------|------------------|---|
| Backlight | bool | True: support backlight |
| ChipID | String | ID of chips in this channel |
| CoverCount | integer | The maximum cover region count. |
| CoverType | integer | 0: don't support cover 1: support realtime cover 2: support non-realtime cover |
| CustomManualExposure | bool, | true: support use defined manual exposure time |
| DayNightColor | bool | true: support color alternate between day and night. |
| DownScaling | bool | true: support down scaling, binning mode not included. |
| Exposure | integer | Exposure grade. 0 –don't support exposure control. |
| ExternalSyncInput | bool | true: support HD signal external synchronization. |
| FlashAdjust | bool | true: support flash adjust |
| Flip | bool | true: support picture flip. |
| Gain | bool | true: support gain control. |
| GainAuto | bool | true: support auto gain. |
| HorizontalBinning | integer | Horizontal/Vertical pixel binning mask, 1 – support 2 pixel binning, 2 – support 3 pixel binning 4 - support 4 pixel binning ... 2^n – support n+2 pixel binning |
| VerticalBinning | integer | |
| InfraRed | bool | true: support Infra compensation |
| Iris | bool | true: support Iris adjust |
| IrisAuto | bool | true: support auto Iris adjust |
| LadenBitrate | integer | Unit is Kbps. Maximum value of video stream bitrate, 16bpp, not in binning mode. |
| LimitedAutoExposure | bool | true: support auto exposure with time limit. |
| MaxHeight | integer | Maximum video height |
| MaxWidth | integer | Maximum video width |

| | | |
|------------------|---------|---|
| Mirror | bool | true: support picture mirror. |
| NightOptions | bool | true: support night options. |
| ReferenceLevel | bool | true: support reference level. |
| Rotate90 | bool | true: support clockwise/anticlockwise 90 ° rotate |
| SetColor | bool | true: support color set. |
| SignalFormats | string | It's a string contains supported video input signal formats for this channel. Signal formats are separated by comma. Range is {Inside, BT656, 720p, 1080p, 1080i, 1080sF, 1_3M} Inside – inside input. 1_3M - 1280*960 |
| SyncChipChannels | bool | True: channels in same chip should be synchronized. Synchronized means video resolution of these channels should be the same. |
| TitleCount | integer | Maximum count of blending titles. |
| UpScaling | bool | true: support up scaling. |
| WhiteBalance | integer | Range is {0, 1, 2, 3} 0 – don't support white balance. 1 – support auto white balance 2 - support auto and pre defined white balance. 3 - support auto, pre defined and user defined white balance |

Example:

/cgi-bin/videoInput.cgi?action=getCaps&channel=0

3.3.2 Set Parameter

Set the parameter of video in options.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]

With the following parameter and value type:

| paramName | ParamValue type | Description |
|-----------|-----------------|-------------|
|-----------|-----------------|-------------|

| | | |
|---------------------------------|---------|---|
| <i>head</i> .Backlight | integer | Range is [0-n] n depends on capability in 3.3.1 GetCapabilities (Video In Options) 0 – backlight closed. 1 – backlight grade 1 ... n – backlight grade n |
| <i>head</i> .DayNightColor | integer | Range is {0,1,3} 0: always multicolor 1:autoswitch along with brightness, 3:always monochrome |
| <i>head</i> .ExposureSpeed | integer | Range is [0-n+1], n depends on capability in 3.3.1 GetCapabilities (Video In Options) 0: AutoExposure 1-n-1:manual Exposure grade n: AutoExposure with time limit. n+1:manualExposure with user-defined time (n issupportedmaximum exposuregrade) |
| <i>head</i> .ExposureValue1 | float | Range is [0.1-80], unit is millisecond If ExposureSpeed is 0(AutoExposure enable), it's lower limit of AutoExposure time, otherwise it's time of manualExposure |
| <i>head</i> .ExposureValue2 | float | Range is[0.1-80], unit is millisecond Upper limit of AutoExposure time, should be bigger than ExposureValue1 |
| <i>head</i> .ExternalSync | integer | Range is {0,1} External Synchronous 0:Internal Synchronization 1: External Synchronous |
| <i>head</i> .ExternalSyncPhase | integer | Range is [0°-360°] External Synchronous Signal Phase |
| <i>head</i> .FlashControl.Mode | integer | Range is {0,1,2} 0:forbid flash 1:always flash 2:auto flash |
| <i>head</i> .FlashControl.Pole | integer | Range is {0,1, 2, 3} Trigger mode: 0:low level 1:high level 2:rising-edge 3:falling-edge |
| <i>head</i> .FlashControl.Value | integer | Range is [0-15] Flashlight time-unit: 0-0us, |

| | | |
|-----------------------------------|---------|--|
| | | 1-64us, 2-128us, 3-192us ... 15-960us |
| head.FlashControl.PreValue | integer | Range is [0-100] It's threshold of brightness value, if brightness is less than this value, flash light begin to work. |
| head.Flip | bool | true: enable video flip function false: disable video flip function |
| head.Gain | integer | Range is [0,1,2,3,4] If GainAuto is true, it's upper limit of auto gain, else it's the fixed as five level with {Low, Lower, Medium, Higher, High}. |
| head.GainBlue | integer | Range is [0-100] Gain for blue value, Value is effective when WhiteBalance is "Custom." |
| head.GainRed | integer | Range is [0-100] Gain for red value, Value is effective when WhiteBalance is "Custom." |
| head.GainGreen | integer | Range is [0-100] Gain for green value, Value is effective when WhiteBalance is "Custom." |
| head.GainAuto | bool | true: GainAuto false: No GainAuto |
| head.IrisAuto | bool | true: IrisAuto false: No IrisAuto |
| head.Mirror | bool | true: enable video mirror function false: disable video mirror function |
| head.WhiteBalance | String | Range is {Disable, Auto, Custom, Sunny, Cloudy, Home, Office, Night} White balance Mode |
| head.ReferenceLevel | integer | Range is [0-100] The expected average brightness level of video frames. |
| head.Rotate90 | integer | Range is {0,1,2} Video rotation: 0: No rotate 1: clockwise rotate 90° 2: anticlockwise rotate 90° |
| head.SignalFormat | String | Range is {Inside, BT656, 720p, 1080p, 1080i, 1080sF} |

| | | Input Signal Mode |
|---|---------|---|
| <i>head</i> .NightOptions.BrightnessThreshold | integer | NightOptions contain a set of parameters used when brightness is not enough. Range is [0-100] when brightness is less than the BrightnessThreshold, parameters change to NightOptions . |
| <i>head</i> .NightOptions.IrisAuto | bool | true: IrisAuto false: No IrisAuto |
| <i>head</i> .NightOptions.SunriseHour | integer | Range is [00-23] Sunrise hour. |
| <i>head</i> .NightOptions.SunriseMinute | integer | Range is [00-59] Sunrise minute |
| <i>head</i> .NightOptions.SunriseSecond | integer | Range is [00-59] Sunrise second |
| <i>head</i> .NightOptions.SunsetHour | integer | Sunset time. Its range is same with sunrise time, and it should be after sunrise time. NightOptions are used if time is after sunset time and before sunrise time. |
| <i>head</i> .NightOptions.SunsetMinute | integer | |
| <i>head</i> .NightOptions.SunsetSecond | integer | |
| <i>head</i> .NightOptions.SwitchMode | integer | Range is {0,1,2} 0:NoSwitch; 1:Switch depends on brightness; 2: Switch depends on time, switch to NightOptions when time is after sunset time and before sunrise. |
| <i>head</i> .NightOptions.ExposureSpeed | integer | Range is same as relevant items of normal options in this table. Example: Value range of <i>head</i> .NightOptions.ExposureSpeed is the same with <i>head</i> .ExposureSpeed |
| <i>head</i> .NightOptions.ExposureValue1 | float | |
| <i>head</i> .NightOptions.ExposureValue2 | float | |
| <i>head</i> .NightOptions.Gain | integer | |
| <i>head</i> .NightOptions.GainAuto | bool | |
| <i>head</i> .NightOptions.GainBlue | integer | |
| <i>head</i> .NightOptions.GainGreen | integer | |
| <i>head</i> .NightOptions.GainRed | integer | |

| | | |
|---|---------|--|
| <i>head</i> .NightOptions.White Balance | String | |
| <i>head</i> .NightOptions.ReferenceLevel | integer | |
| <i>head</i> .NightOptions.ExternalSyncPhase | integer | |

comment:

In above table, *head*=VideoInOptions[*ChannelNo*]
ChannelNo = video channel index.

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&VideoInOptions[0].Backlight=0

3.3.3 Get Parameter

Video in options contain Backlight, ExposureSpeed, DayNightColor, NightOptions and so on.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=**VideoInOptions**

Response:

head.Backlight=0
head.DayNightColor=false
head.ExposureSpeed=0
head.ExposureValue1=0.100000
head.ExposureValue2=80.000000
head.ExternalSync=0
head.ExternalSyncPhase=0
head.FlashControl.Mode=0
head.FlashControl.Pole=0
head.FlashControl.Value=0
head.FlashControl.PreValue=0
head.Flip=false
head.Gain=50

```
head.GainAuto=true
head.IrisAuto=false
head.Mirror=false
head.NightOptions.BrightnessThreshold=50
head.NightOptions.ExposureSpeed=0
head.NightOptions.ExposureValue1=0.100000
head.NightOptions.ExposureValue2=80
head.NightOptions.Gain=50
head.NightOptions.GainAuto=true
head.NightOptions.GainBlue=50
head.NightOptions.GainGreen=50
head.NightOptions.GainRed=50
head.NightOptions.IrisAuto=false
head.NightOptions.SunriseHour=0
head.NightOptions.SunriseMinute=0
head.NightOptions.SunriseSecond=0
head.NightOptions.SunsetHour=0
head.NightOptions.SunsetMinute=0
head.NightOptions.SunsetSecond=0
head.NightOptions.SwitchMode=0
head.NightOptions.WhiteBalance=Disable
head.ReferenceLevel=50
head.ReferenceLevelEnable=false
head.Rotate90=0
head.SignalFormat=BT656
head.WhiteBalance=Disable
```

comment:

In above table, **head**=table.VideoInOptions[*ChannelNo*]
ChannelNo = video channel index.

3.4 Video Encode

3.4.1 Get Capabilities

Get video configcapabilities.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/encode.cgi?action=getParamCaps
```

Response:

```

headMain.Video.BitRateOptions=2048,12288
headMain.Video.CompressionTypes=H.264
headMain.Video.FPSMax=25
headMain.Video.ResolutionTypes= 1280 x 960,720,D1
headExtra.Video.BitRateOptions=768,4096
headExtra.Video.CompressionTypes=H.264
headExtra.Video.FPSMax=25
headExtra.Video.ResolutionTypes=D1,CIF
headSnap.Video.CompressionTypes=H.264
headSnap.Video.ResolutionTypes=1280 x 960,720,D1,CIF
    
```

With the following parameter and value type:

| Field In Respos | ParamValue type | Description |
|------------------|-----------------|---|
| BitRateOptions | string | Before comma is minimum bit rate. (kbps), after comma is maximum bit rate.(kbps) BitRateOptions=80,448 80 is minimum bitrate, 448 is maximum. |
| CompressionTypes | string | It contains all supported video compression types separated by comma. Range is {MPEG4, MPEG2, MPEG1, MJPG, H.263, H.264} |
| FPSMax | integer | Maximum FPS. |
| ResolutionTypes | string | It contains all supported video resolutions. Range is in 3.4.2 Resolution . |

comment:

In above table:

Channel: video channel index

RecordType:

- 0 = regular record
- 1 = motion detection record
- 2 = alarm record

ExtraStream:

- 0 = extra stream 1
- 1 = extra stream 2
- 2 = extra stream 3

SnapType:

- 0 = regular snapshot
- 1 = motion detection snapshot
- 2 = alarm snapshot

Abbreviations in above table:

headMain = caps[Channel].MainFormat[RecordType]
headExtra = caps[Channel].ExtraFormat[ExtraStream]
headSnap = caps[Channel].SnapFormat[SnapType]

3.4.2 Resolution

| Fixed Resolution Name | Size in PAL | Size in NTSC |
|-----------------------|-------------------------------|--------------|
| "D1" | 704 x 576 | 704 x 480 |
| "HD1" | 352 x 576 | 352 x 480 |
| "BCIF" | 704 x 288 | 704 x 240 |
| "CIF" | 352 x 288 | 352 x 240 |
| "QCIF" | 176 x 144 | 176 x 120 |
| "VGA" | 640 x 480 | |
| "QVGA" | 320 x 240 | |
| "SVCD" | 480 x 480 | |
| "QQVGA" | 160 x 128 | |
| "SVGA" | 800 x 592 | |
| "XVGA" | 1024 x 768 | |
| "WXGA" | 1280 x 800 | |
| "SXGA" | 1280 x 1024 | |
| "WSXGA" | 1600 x 1024 | |
| "UXGA" | 1600 x 1200 | |
| "WUXGA" | 1920 x 1200 | |
| "ND1" | 240 x 192 | |
| "720" | 1280 x 720 | |
| "1080" | 1920 x 1080 | |
| "1280x960" | 1280 x 960 (1.3 Mega Pixels) | |
| "1872x1408" | 1872 x 1408 (2.5 Mega Pixels) | |
| "3744x1408" | 3744 x 1408 (5 Mega Pixels) | |
| "2048x1536" | 2048 x 1536 (3 Mega Pixels) | |
| "2432x2048" | 2432 x 2048 (5 Mega Pixels) | |
| "1216x1024" | 1216 x 1024 (1.2 Mega Pixels) | |
| "1408x1024" | 1408 x 1024 (1.5 Mega Pixels) | |
| "3296x2472" | 3296 x 2472 (8 Mega Pixels) | |
| "2560x1920" | 2560 x 1920 (5 Mega Pixels) | |
| "960H", | 960 x 576 | 960 x 480 |
| "DV720P" | 960 x 720 | |

3.4.3 Set Parameter

Set the parameter of the video encoder.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | param Value type | Description |
|----------------------------------|------------------|--|
| <i>head.Video.BitRate</i> | integer | Unit is Kbps Range depends on capability in 3.4.1 GetCapabilites (Video Encode) |
| <i>head.Video.BitRateControl</i> | string | Range is { CBR,VBR } CBR:constant bitrate VBR:variable bitrate |
| <i>head.Video.Compression</i> | String | Range is {MPEG4,MPEG2,MPEG1,MJPEG,H.263,H.264} Depends on capacity in 3.4.1 GetCapabilites (Video Encode) |
| <i>head.Video.FPS</i> | float | Range is [1-30]. Frames per second. < 1.0: several seconds/frame, FPS=0.3333: 3 seconds per frame. >1.0: several frames/second. FPS=3: 3 frames per second. |
| <i>head.Video.GOP</i> | integer | Range is [1-150]. Group of picture, it's the interval of IFrame, Example: GOP=50, means there is one I frame every 49 P or B frames |
| <i>head.Video.Height</i> | integer | Video height |
| <i>head.Video.Width</i> | integer | Video Width |
| <i>head.Video.Profile</i> | String | Range is { Baseline, Main , Extended , High } Only when video compression is H.264, it's effective. |
| <i>head.Video.Quality</i> | integer | Range is [1-6]. Image Quality, available when Video.BitRateControl= VBR 1: worst quality |

| | | |
|-------------------------|------|--------------------|
| | | 6: best quality |
| <i>head.VideoEnable</i> | bool | True: enable video |

comment:

Channel: video channel index

RecordType:

- 0 = regular record
- 1 = motion detection record
- 2 = alarm record

ExtraStream:

- 0 = extra stream 1
- 1 = extra stream 2
- 2 = extra stream 3

Abbreviation in above table:

head=Encode[*Channel*].MainFormat[*RecordType*] (or)

Encode[*Channel*].ExtraFormat[*ExtraStream*]

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&Encode[0].ExtraFormat[1].Video.GOP=50

3.4.4 Get Parameter

Get the video encode parameter.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=**Encode**

Response:

headMain.Video.BitRate=8192

headMain.Video.BitRateControl=CBR

headMain.Video.Compression=H.264

headMain.Video.FPS=25

headMain.Video.GOP=50

headMain.Video.Height=1200

headMain.Video.Profile=Main

```
headMain.Video.Quality=4  
headMain.Video.Width=1600  
headMain.Video.Enable=true  
headExtra.Video.BitRate=8192  
headExtra.Video.BitRateControl=CBR  
headExtra.Video.Compression=H.264  
headExtra.Video.FPS=25  
headExtra.Video.GOP=50  
headExtra.Video.Height=1200  
headExtra.Video.Profile=Main  
headExtra.Video.Quality=4  
headExtra.Video.Width=1600  
headExtra.Video.Enable=true
```

comment:

Channel: video channel index

RecordType:

- 0 = regular record
- 1 = motion detection record
- 2 = alarm record

ExtraStream:

- 0 = extra stream 1
- 1 = extra stream 2
- 2 = extra stream 3

Abbreviations in above table:

headMain= table.Encode[*Channel*].MainFormat[*RecordType*]

headExtra =table.Encode[*Channel*].ExtraFormat[*ExtraStream*]s

3.5 Audio Encode

3.5.1 Get Capabilities

Get audioconfigcapabilities.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/encode.cgi?action=getParamCaps
```

Response:

```
caps[0].ExtraFormat[0].Audio.CompressionTypes=PCM,G.711A,G.711Mu
caps[0].ExtraFormat[1]...
...
caps[0].MainFormat[0].Audio.CompressionTypes=PCM,G.711A,G.711Mu
caps[0].MainFormat[1]...
...
```

With the following parameter and value type:

| Field In Respos | ParamValue type | Description |
|------------------|-----------------|---|
| CompressionTypes | string | It contains all supported audio compression types, separated by comma. Range is {PCM,ADPCM,G.711A,G.711Mu,G.726,G.729,MPEG2,AMR} |

comment:

The angle brackets above denotes a array

3.5.2 Set Parameter

Set the audio encode parameter.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|--------------------------------|-----------------|---|
| <i>head</i> .Audio.Bitrate | integer | Unit is kbps Range depends on capacity in 3.5.1 GetCapabilites (Audio Encode) |
| <i>head</i> .Audio.Compression | string | Range depends on capacity in 3.5.1 GetCapabilites (Audio Encode) |
| <i>head</i> .Audio.Depth | integer | Audio sampling depth |
| <i>head</i> .Audio.Frequency | integer | Audio sampling frequency |
| <i>head</i> .Audio.Mode | integer | Range is {0,1,2,3,4,5,6,7} Audio encode mode. 0: 4.75kbps, 1: 5.15 kbps, 2: 5.9 kbps, |

| | | |
|--------------------------|------|---|
| | | 3: 6.7 kbps, 4: 7.4 kbps, 5: 7.95 kbps, 6: 10.2 kbps, 7: 12.2 kbps, |
| <i>head</i> .AudioEnable | bool | Enable/Disableaudio |

comment:

Channel: video channel index

RecordType:

0 = regular record

1 = motion detection record

2 = alarm record

ExtraStream:

0 = extra stream 1

1 = extra stream 2

2 = extra stream 3

Abbreviations in above table:

head=Encode[*Channel*].MainFormat[*RecordType*] (or)

Encode[*Channel*].ExtraFormat[*ExtraStream*]

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&Encode[0].ExtraFormat[1].Audio.Bitrate=64

3.5.3 Get Parameter

Get the audio encode parameter.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=**Encode**

Response:

headMain.Audio.Bitrate=64

headMain.Audio.Compression=G.711A

```
headMain.Audio.Depth=16  
headMain.Audio.Frequency=44000  
headMain.Audio.Mode=0  
headMain.Audio.Enable=false  
headExtra.Audio.Bitrate=64  
headExtra.Audio.Compression=G.711A  
headExtra.Audio.Depth=16  
headExtra.Audio.Frequency=44000  
headExtra.Audio.Mode=0  
headExtra.Audio.Enable=false
```

comment:

Channel: video channel index

RecordType:

- 0 = regular record
- 1 = motion detection record
- 2 = alarm record

ExtraStream:

- 0 = extra stream 1
- 1 = extra stream 2
- 2 = extra stream 3

Abbreviations in above table:

headMain=table.Encode[*Channel*].MainFormat[*RecordType*]

headExtra=table.Encode[*Channel*].ExtraFormat[*ExtraStream*]

3.6 Snap Encode

3.6.1 Get Capabilities

Get snap configcapabilities.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/encode.cgi?action=getParamCaps
```

Response:

```
caps[Channel].SnapFormat[SnapType].Video.CompressionTypes=H.264
```

caps[*Channel*].SnapFormat[*SnapType*].Video.ResolutionTypes=1280x960,720,D1,CIF

With the following parameter and value type:

| Field In Responses | Param Value type | Description |
|--------------------|------------------|---|
| Compression Types | string | It contains all supported video compression types separated by comma. Range is {MPEG4, MPEG2, MPEG1, MJPG, H.263, H.264} |
| Resolution Types | string | It contains all supported video resolutions, separated by comma. Range is {D1,HD1,BCIF,CIF,QCIF,VGA,QVGA,SVGA,XVGA,WXGA,SXGA,WSXGA,UXGA,WUXGA,ND1,720,1080,1_3M,2_5M,3M,5M}. |

comment:

Channel: video channel index

SnapType:

0 = regular snapshot

1 = motion detection snapshot

2 = alarm snapshot

3.6.2 Set Parameter

Set the snap encode parameter.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]

With the following parameter and value type:

| paramName | paramValue type | Description |
|----------------------------------|-----------------|---|
| <i>head.Video.BitRate</i> | integer | Unit is Kbps Range depends on capability in 3.3.1 GetCapabilities (Video In Options) |
| <i>head.Video.BitRateControl</i> | string | Range is {CBR,VBR} CBR:constant bitrate |

| | | |
|-------------------------------|---------|--|
| | | VBR:variable bitrate |
| <i>head.Video.Compression</i> | String | Range is {MPEG4,MPEG2, MPEG1,MJPEG,H.263,H.264} Depends on capacity in 3.3.1 GetCapabilities(Video In Options) |
| <i>head.Video.FPS</i> | float | Range is [0.2-30]. Frames per second. < 1.0: several seconds/frame, FPS=0.3333: 3 seconds per frame. >1.0: several frames/second. FPS=3: 3 frames per second. |
| <i>head.Video.GOP</i> | integer | Range is [1-100]. Group of picture, it's the interval of IFrame, Example: GOP=50, means there is one I frame every 49 P or B frames |
| <i>head.Video.Height</i> | integer | Video height |
| <i>head.Video.Width</i> | integer | Video Width |
| <i>head.Video.Quality</i> | integer | Range is [1-6]. Image Quality, available when Video.BitRateControl=VBR 1: worst quality 6: best quality |
| <i>head.VideoEnable</i> | bool | True: enable video |

comment:

Channel: video channel index

SnapType:

- 0 = regular snapshot
- 1 = motion detection snapshot
- 2 = alarm snapshot

Abbreviation in above table:

head= Encode[**Channel**].SnapFormat[**SnapType**]

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&Encode[0].SnapFormat[0].AudioEnable=False

3.6.3 Get Parameter

Get the snap encode parameter.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=Encode [Channel].SnapFormat
```

Response:

```
headSnap.Video.BitRate=384  
headSnap.Video.BitRateControl=VBR  
headSnap.Video.Compression=H.264  
headSnap.Video.FPS=1  
headSnap.Video.GOP=50  
headSnap.Video.Height=576  
headSnap.Video.Quality=4  
headSnap.Video.Width=704  
headSnap.VideoEnable=true
```

comment:

Channel: video channel index

SnapType:

- 0 = regular snapshot
- 1 = motion detection snapshot
- 2 = alarm snapshot

Abbreviations in above table:

headSnap = table.Encode[*Channel*].SnapFormat[*SnapType*]

Example:

```
/cgi-bin/paramManager.cgi?action=getParam&name=Encode[0].SnapFormat
```

3.7 Channel Title

3.7.1 Set Parameter

Set the channel title parameter.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam<paramName>=<paramValue>
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|-------------------------------------|-----------------|--------------|
| ChannelTitle[<i>Channel</i>].Name | String | Channel Name |

comment:

Set the title of the channel.

If VideoWidget[*Channel*].ChannelTitle.EncodeBlend is true, this title is blended to the video frames.

Please refer to [3.9.1 SetParam \(Video Widget\)](#)

In above table, *Channel*: video channel index

Response:

OK or ERROR

Example:

```
/cgi-bin/paramManager.cgi?action=setParam&ChannelTitle[0].Name=ABC1
```

3.7.2 Get Parameter

Get the channel title parameter.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=ChannelTitle
```

Response:

```
table.ChannelTitle[Channel].Name=CAM1
```

comment:

Get the title of the channel.
In above table, *Channel*= video channel index

3.8 Video Standard

3.8.1 Get Parameter

Get the standard parameter in this IPC.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=VideoStandard
```

Response:

```
table.VideoStandard=PAL
```

comment:

The standard of video in {PAL,NTSC}

3.9 Video Widget

3.9.1 Set Parameter

Set the video widget parameter.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|--|------------------------|--|
| <i>headCover</i> .BackColor[0] <i>headCover</i> .BackColor[1] <i>headCover</i> .BackColor[2] <i>headCover</i> .BackColor[3] | integer | Range is [0-255]. BackColor[0]:red value BackColor[1]:green value BackColor[2]:blue value BackColor[3]: alpha value |
| <i>headCover</i> .EncodeBlend | bool | false - widget blend is disabled. |
| <i>headCover</i> .FrontColor[0] <i>headCover</i> .FrontColor[1] <i>headCover</i> .FrontColor[2] <i>headCover</i> .FrontColor[3] | integer | Range is [0-255]. FrontColor[0]:red value FrontColor[1]:green value FrontColor[2]:blue value FrontColor[3]: alpha value |
| <i>headCover</i> .Rect[0] <i>headCover</i> .Rect[1] <i>headCover</i> .Rect[2] <i>headCover</i> .Rect[3] | integer | Range is [0-8191]. Rect[0]: top left corner x coordinate (left) Rect[1]: top left corner y coordinate (top) Rect[2]: bottom right x coordinate (right) Rect[3]: bottom right y coordinate (bottom) |
| <i>headChannelTitle</i> .BackColor[0] <i>headChannelTitle</i> .BackColor[1] <i>headChannelTitle</i> .BackColor[2] <i>headChannelTitle</i> .BackColor[3] | integer | Range is the same with <i>headCover</i> |
| <i>headChannelTitle</i> .EncodeBlend | Bool | |
| <i>headChannelTitle</i> .FrontColor[0] <i>headChannelTitle</i> .FrontColor[1] <i>headChannelTitle</i> .FrontColor[2] <i>headChannelTitle</i> .FrontColor[3] | Integer | |
| <i>headChannelTitle</i> .Rect[0] <i>headChannelTitle</i> .Rect[1] <i>headChannelTitle</i> .Rect[2] <i>headChannelTitle</i> .Rect[3] | integer | Only use the value of (left,top),the value of (right,bottom) is the same as (left,top) Rect[0], Rect[1] are used, and Rect[2] must be same with Rect[0], Rect[3] must be same with Rect[1]. |
| <i>headTimeTitle</i> .BackColor[0] <i>headTimeTitle</i> .BackColor[1] <i>headTimeTitle</i> .BackColor[2] <i>headTimeTitle</i> .BackColor[3] | integer | Range is the same with <i>headChannelTitle</i> These are params about time title. |
| <i>headTimeTitle</i> .EncodeBlend | bool | |
| <i>headTimeTitle</i> .FrontColor[0] <i>headTimeTitle</i> .FrontColor[1] <i>headTimeTitle</i> .FrontColor[2] <i>headTimeTitle</i> .FrontColor[3] | integer | |
| <i>headTimeTitle</i> .Rect[0] <i>headTimeTitle</i> .Rect[1] | integer | |

| | | |
|--------------------------------|------|---|
| <i>headTimeTitle</i> .Rect[2] | | |
| <i>headTimeTitle</i> .Rect[3] | | |
| <i>headTimeTitle</i> .ShowWeek | bool | True: Display week within the time title. |

comment:

Channel: video channel index
 CoReg :Cover region index
 Covers is an array which contains multiple cover regions
 0 = region 1
 1 = region 2
 2 = region 3
 3 = region 4
headChannelTitle=VideoWidget[*Channel*].ChannelTitle
headCover = VideoWidget[*Channel*].Covers[*CoReg*]
headTimeTitle = VideoWidget[*Channel*].TimeTitle

VideoWidgetparam contains cover region settings, channel title settings and time title settings.
 The italics above will be replaced by the above abbreviations.

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&VideoWidget[0].ChannelTitle.BackColor[0]=0&VideoWidget[0].ChannelTitle.BackColor[1]=0&VideoWidget[0].ChannelTitle.BackColor[2]=0&VideoWidget[0].ChannelTitle.BackColor[3]=128

3.9.2 Get Parameter

VideoWidgetparam contains ChannelTitle, Covers and TimeTitle parameters, defines the background color, front color and positions of channel title and time title, and defines the regions which are not visible (cover).

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=**VideoWidget**

Response:

head.BackColor[0]=0

```
head.BackColor[1]=0  
head.BackColor[2]=0  
head.BackColor[3]=128  
head.EncodeBlend=true  
head.FrontColor[0]=255  
head.FrontColor[1]=255  
head.FrontColor[2]=255  
head.FrontColor[3]=0  
head.PreviewBlend=true  
head.Rect[0]=0  
head.Rect[1]=8191  
head.Rect[2]=0  
head.Rect[3]=8191  
...  
...
```

comment:

```
Channel: video channel index  
CoReg: Cover Region  
  Covers is an array which sustains multi- Cover regions  
    0 = region 1  
    1 = region 2  
    2 = region 3  
    3 = region 4  
head=table.VideoWidget[Channel].ChannelTitle (or)  
  table.VideoWidget[Channel].Covers[CoReg](or)  
table.VideoWidget[Channel].TimeTitle
```

4 VIDEO RECORD

4.1 Record

4.1.1 Set Parameter

Set the channel number or the timesection in record parameters.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]

With the following parameter and value type:

| paramName | paramValue type | Description |
|---|-----------------|--|
| Record[<i>ch</i>].PreRecord | integer | Range is [0-300]. Prerecord seconds, 0 means no prerecord. <i>ch</i> (Channel number) starts form 0 |
| Record[<i>ch</i>].TimeSection[<i>wd</i>][<i>ts</i>] | string | <i>wd</i> (week day) range is [0-6] (Sunday -Staurday) <i>ts</i> (time section) range is [0-23], timesection table index. Format: mask hh:mm:ss-hh:mm:ss Mask: [0-65535], hh: [0-24], mm: [0-59], ss: [0-59] Mask indicates record type by bits: Bit0: regular record Bit1: motion detection record Bit2: alarm record Bit3: card record |

comment:

In above table:
ch = channel index,
wd = week day index,
ts = time section index

Response:

OK or ERROR

Example:

Set record time to every Sunday all day. Record type is motion detection and alarm.

URL should be:

`http://<ip>/cgi-bin/paramManager.cgi?action=setParam&name=Record[0].TimeSection[0][0]&table=6 00:00:00-24:00:00`

In this example, “6 00:00:00-24:00:00” means motion detection and alarm record all day (6 = 4 & 2, alarm is 4, motion detection is 2.).

4.1.2 Get Parameter

Get the parameter of record.

Method: GET

Syntax:

`http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=Record`

Response:

```
table.Record[channel].PreRecord=6  
table.Record[channel].TimeSection[weekday][0]=1 00:00:00-24:00:00  
table.Record[channel].TimeSection[weekday][1]=0 02:00:00-24:00:00  
table.Record[channel].TimeSection[weekday][2]=0 03:00:00-24:00:00  
table.Record[channel].TimeSection[weekday][3]=0 04:00:00-24:00:00  
table.Record[channel].TimeSection[weekday][4]=0 05:00:00-24:00:00  
table.Record[channel].TimeSection[weekday][5]=0 06:00:00-24:00:00
```

comment:

Channel in above table is video channel number, *weekday* range is [0-6] (Sunday - Saturday).
Record param contains pre record time and record time sections of every day.

4.1.3 Set Mode Parameter

Set the mode of record parameter.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=setParam<paramName>=<paramValue>[&<paramName>=<paramValue>...]

With the following parameter and value type:

| paramName | paramValue type | Description |
|-----------------------------------|-----------------|---|
| RecordMode[<i>channel</i>].Mode | Integer | Range is {0, 1, 2}. 0: automatically record 1: manually record 2: stop record. |

comment:

channel in above table is video channel index, start form 0.

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&RecordMode[0].Mode=1

4.1.4 Get Mode Parameter

Get the parameter of record mode.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=**RecordMode**

Response:

table.RecordMode[*channel*].Mode=0

comment:

Get record mode for video channels. *channel* in above table is video channel number.

4.2 Snap

4.2.1 Set Parameter

Set the snap parameter.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|---|-----------------|--|
| Record[<i>ch</i>].TimeSection[<i>wd</i>][<i>ts</i>] | string | <p>wd (week day) range is [0-6] (Sunday-Staurday)</p> <p>ts (time section) range is [0-23], it's timesection table index.</p> <p>Format: mask hh:mm:ss-hh:mm:ss Mask: [0-65535], hh: [0-24], mm: [0-59], ss: [0-59]</p> <p>Mask indicates record type by bits: Bit0: regular snapshot Bit1: motion detection snapshot Bit2: alarm snapshot Bit3: card snapshot</p> |

comment:

In above table:
ch = channel index,
wd = week day index,
ts = time section index

Response:

OK or ERROR

Example:

```
/cgi-bin/paramManager.cgi?action=setParam&Snap[0].TimeSection[0][0]=6 00:00:00-23:59:59
```

4.2.2 Get parameter

Get the parameter of snap.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=Snap
```

Response:

```
table.Snap[channel].TimeSection[weekday][0]=1 00:00:00-24:00:00  
table.Snap[channel].TimeSection[weekday][1]=0 02:00:00-24:00:00  
table.Snap[channel].TimeSection[weekday][2]=0 03:00:00-24:00:00  
table.Snap[channel].TimeSection[weekday][3]=0 04:00:00-24:00:00  
table.Snap[channel].TimeSection[weekday][4]=0 05:00:00-24:00:00  
table.Snap[channel].TimeSection[weekday][5]=0 06:00:00-24:00:00
```

comment:

Channel in above table is video channel number, *weekday* range is [0-6] (Sunday - Saturday).

4.2.3 Snap Picture

Get one snap picture.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/snapshot.cgi?action=Enable
```

Response:

With one snap picture.

5 NETWORK

5.1 Net Interface

5.1.1 Get Interface

Get the net interface infomations.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/net.cgi?action=getInterfaces
```

Response:

```
netInterface[0].Name=eth0  
netInterface[0].Type=Normal  
netInterface[0].Valid=true  
netInterface[1]....  
...
```

comment:

Get all of the system network interfaces.

Description for items In above table

Name: network interface name.

“eth0” - wired network interface

“eth2” - wireless network interface

“3G” - 3G network interface

Type: “Normal”– wired network

“Wireless”– wireless network

"Auto", "TD-SCDMA", "WCDMA", "CDMA1x", "EDGE", "EVDO"– 3G network types.

Valid: network interface is valid if netInterface[n].Valid is true.

5.2 Basic Parameter

5.2.1 Set Parameter

Set the basic parameter of network.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|--|-----------------|--|
| NetWork.DefaultInterface | string | Set default network interface when multiple interfaces exist. Range of interfaces is depends on 5.1.1 GetInterfaces (Net Interface) |
| NetWork.Domain | string | Domain name. |
| NetWork.Hostname | string | Hostname and Domain compose a network address. |
| Network. <i>interface</i> .DefaultGateway | string | IP address |
| Network. <i>interface</i> .DhcpEnable | bool | Enable/Disable DHCP. |
| Network. <i>interface</i> .DnsServers[0] | string | IP address of first DNS server. |
| Network. <i>interface</i> .DnsServers[1] | string | IP address of second DNS server. |
| Network. <i>interface</i> .IPAddress | string | Interface IP address. |
| Network. <i>interface</i> .MTU | integer | Interface MTU. |
| Network. <i>interface</i> .PhysicalAddress | string | MAC address of interface. HEX string in the form of: xx:xx:xx:xx:xx:xx. Range of x is [0-9,a-f,A-F] Example: 00:10:5c:f2:1c:b4 00:10:5C:F2:1C:B5 |
| Network. <i>interface</i> .SubnetMask | string | Network mask string: In the form of x.x.x.x, range of x is [0-255] Example: 255.255.255.0 |

comment:

interface in above table is network interface name, such as eth0, eth1...

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&Network.Hostname=IPC

5.2.2 Get Parameter

Get the basic parameter of network.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=**Network**

Response:

```
table.Network.DefaultInterface=eth0
table.Network.Domain=ya
table.Network.Hostname=badak
table.Network.interface.DefaultGateway=10.7.0.1
table.Network.interface.DhcpEnable=false
table.Network.interface.DnsServers[0]=221.123.33.228
table.Network.interface.DnsServers[1]=221.12.1.228
table.Network.interface.IPAddress=10.7.2.3
table.Network.interface.MTU=1500
table.Network.interface.PhysicalAddress=00:10:5c:f2:1c:b4
table.Network.interface.SubnetMask=255.255.0.0
```

comment:

BasicParam contains basic network parameters (Default interface, domain name, host name), and configuration of each network interface.

interface in above table is network interface name, such as eth0, eth2...

5.3 PPPoE

5.3.1 Set Parameter

Set the parameter of PPPoE.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|----------------|-----------------|-----------------------|
| PPPoE.Enable | bool | Enable/Disable PPPoE. |
| PPPoE.UserName | string | PPPoE user name. |
| PPPoE.Password | string | PPPoE user password. |

Response:

OK or ERROR

Example:

```
/cgi-bin/paramManager.cgi?action=setParam&PPPoE.UserName=admin
```

5.3.2 Get Parameter

Get the parameter of PPPoE.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=PPPoE
```

Response:

```
table.PPPoE.Enable=false  
table.PPPoE.Password=123456  
table.PPPoE.UserName=123456
```

5.4 DDNS

5.4.1 Set Parameter

Set the parameter of DDNS.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]

With the following parameter and value type:

| paramName | paramValue type | Description |
|--------------------------------|-----------------|---|
| DDNS[<i>index</i>].Address | string | DDNS server IP address or name. |
| DDNS[<i>index</i>].Enable | bool | Multiple DDNS hostname can be set, but Only one hostname can be enabled, others should be disabled. |
| DDNS[<i>index</i>].HostName | String | Host name of this device. |
| DDNS[<i>index</i>].KeepAlive | integer | Range is [1-65535]. Unit is minutes. |
| DDNS[<i>index</i>].Password | string | DDNS user password |
| DDNS[<i>index</i>].Port | integer | Range is [1-65535]. Port of DDSN server |
| DDNS[<i>index</i>].Protocol | string | Range is {NO-IP DDNS, Dyndns DDNS, ZENO}. DDSN protocol type |
| DDNS[<i>index</i>].UserName | string | DDNS user name |

comment:

Index above is the DDNS protocol table index, start from 0.

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&DDNS[0].Address=dynupdate.no-ip.com

5.4.2 Get Parameter

Get the DDNS parameters.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=Email
```

Response:

```
table.Email.Address=www.google.com  
table.Email.Anonymous=true  
table.Email.AttachEnable=true  
table.Email.AttachmentEnable=true  
table.Email.Enable=true  
table.Email.HealthReport.Enable=false  
table.Email.HealthReport.Interval=61  
table.Email.Password=123456  
table.Email.Port=26  
table.Email.Receivers[0]=x@inesa-e.com  
table.Email.Receivers[1]=y@inesa-e.com  
table.Email.Receivers[2]=z@inesa-e.com  
table.Email.SendAddress=x@inesa-e.com  
table.Email.SslEnable=false  
table.Email.Title=DVRMessage  
table.Email.UserName=anonymitty
```

5.5 UPNP

5.5.1 Set Parameter

Set the parameter of UPNP.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|---|-----------------|---|
| UPnP.Enable | bool | Enable/Disable UPNP feature. |
| UPnP.MapTable[<i>index</i>].Enable | bool | Enable/Disable this UPNP map. |
| UPnP.MapTable[<i>index</i>].InnerPort | integer | Range is [1-65535]. Inner port number |
| UPnP.MapTable[<i>index</i>].OuterPort | integer | Range is [1-65535]. Outer port number. |
| UPnP.MapTable[<i>index</i>].Protocol | string | Range is {TCP, UDP} |
| UPnP.MapTable[<i>index</i>].ServiceName | string | User defined UPnP service name. |

comment:

Index in above table is UPnP map table index, range is [0-255]

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&UPnP.MapTable[1].InnerPort=8000

5.5.2 Get Parameter

Get the parameter of UPNP.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=UPnP

Response:

table.UPnP.Enable=true
table.UPnP.MapTable[*index*].Enable=true
table.UPnP.MapTable[*index*].InnerPort=80
table.UPnP.MapTable[*index*].OuterPort=8080
table.UPnP.MapTable[*index*].Protocol=TCP
table.UPnP.MapTable[*index*].ServiceName=HTTP

comment:

Index in above is the UPNP map table index, start from 0.s

5.5.3 Get State

Get the state of UPNP.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/net.cgi?action=getUPnPStatus
```

Response:

```
result=1
```

comment:

```
Get UPNP mapping result:
result=1: mapping succeed.
result=0: mapping failed.
```

5.6 NTP

5.6.1 Set Parameter

Set the parameter of NTP.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|--------------|-----------------|--|
| NTP.Address | string | NTP server IP address or name. |
| NTP.Enable | bool | Enable/Disable NTP server. |
| NTP.Port | integer | Range is [1-65535]. Port of NTP server. |
| NTP.TimeZone | integer | Range is [0-32]. 0: "GMT+00:00" 1: "GMT+01:00" 2: "GMT+02:00" |

| | | |
|------------------|---------|---|
| | | 3: "GMT+03:00" 4: "GMT+03:30" 5: "GMT+04:00" 6: "GMT+04:30" 7: "GMT+05:00" 8: "GMT+05:30" 9: "GMT+05:45" 10: "GMT+06:00" 11: "GMT+06:30" 12: "GMT+07:00" 13: "GMT+08:00" 14: "GMT+09:00" 15: "GMT+09:30" 16: "GMT+10:00" 17: "GMT+11:00" 18: "GMT+12:00" 19: "GMT+13:00" 20: "GMT-01:00" 21: "GMT-02:00" 22: "GMT-03:00" 23: "GMT-03:30" 24: "GMT-04:00" 25: "GMT-05:00" 26: "GMT-06:00" 27: "GMT-07:00" 28: "GMT-08:00" 29: "GMT-09:00" 30: "GMT-10:00" 31: "GMT-11:00" 32: "GMT-12:00" |
| NTP.UpdatePeriod | integer | Range is [0-65535], unit is minutes |

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&NTP.Port=123

5.6.2 Get Parameter

Get the parameter of NTP.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=NTP
```

Response:

```
table.NTP.Address=clock.isc.org
table.NTP.Enable=false
table.NTP.Port=38
table.NTP.TimeZone=9
table.NTP.UpdatePeriod=31
```

5.7 Alarm Server

5.7.1 Set Parameter

Set the parameter of alarm server.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|---------------------|-----------------|--|
| AlarmServer.Address | string | Alarm server IP address or name. |
| AlarmServer.Enable | bool | Enable/Disable Alarm server. |
| AlarmServer.Port | integer | Range is [1-65535]. Port of Alarm server. |

Response:

```
OK or ERROR
```

Example:

```
/cgi-bin/paramManager.cgi?action=setParam&AlarmServer.Port=8888
```

5.7.2 Get Parameter

Get the parameter of alarm server.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=AlarmServer
```

Response:

```
table.AlarmServer.Address=0.0.0.0  
table.AlarmServer.Enable=true  
table.AlarmServer.Port=37777
```

6 EVENT

6.1 Event Handler

6.1.1 Set Parameter

Set the parameter of EventHandler.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]

With the following parameter and value type:

| paramName | paramValue type | Description |
|--|-----------------|---|
| <i>handlerName</i> .EventHandler.AlarmOutChannels[<i>ch</i>] | integer | Range is {0, 1}, <i>ch</i> is alarm out channel index. 0 – do not output alarm at alarm out channel <i>ch</i> 1 – output alarm at alarm out channel <i>ch</i> |
| <i>handlerName</i> .EventHandler.AlarmOutEnable | bool | Enable/Disable alarm out function. |
| <i>handlerName</i> .EventHandler.AlarmOutLatch | Integer | Range is [10-300]. Unit is seconds, indicates the time to output alarm after input alarm is cleared. |
| <i>handlerName</i> .EventHandler.BeepEnable | bool | Enable/Disable beep. |
| <i>handlerName</i> .EventHandler.Dejitter | integer | Range is [0-255]. Alarm signal dejitter seconds. Alarm signal change during this period is ignored. |
| <i>handlerName</i> .EventHandler.Delay | integer | Range is [0-300]. Delay seconds before setting take effect. |
| <i>handlerName</i> .EventHandler.LogEnable | bool | Enable/Disable log for |

| | | |
|--|---------|---|
| | | alarm. |
| <i>handlerName</i> .EventHandler.MailEnable | bool | Enable/Disable mail send for alarm. |
| <i>handlerName</i> .EventHandler.PtzLink[<i>ch</i>][0] | string | Range is {None, Preset, Tour, Pattern} This is PTZ action linked with events. <i>ch</i> is PTZ channel index. |
| <i>handlerName</i> .EventHandler.PtzLink[<i>ch</i>][1] | integer | This is the parameter of PtzLink[<i>ch</i>][0], If PtzLink[<i>ch</i>][0] is Preset: this is preset point. Tour: this is tour path number. Pattern: this is pattern number. |
| <i>handlerName</i> .EventHandler.PtzLinkEnable | Bool | Enable/Disable PTZ link. |
| <i>handlerName</i> .EventHandler.RecordChannels[<i>ch</i>] | Integer | Range is {0, 1} 0 – do not record on video channel <i>ch</i> 1 – record. on video channel <i>ch</i> |
| <i>handlerName</i> .EventHandler.RecordEnable | bool | Enable/Disable record function. |
| <i>handlerName</i> .EventHandler.RecordLatch | integer | Range is [10-300]. Unit is seconds, indicates the time to record after input alarm is cleared.. |
| <i>handlerName</i> .EventHandler.SnapshotChannels[<i>ch</i>] | integer | Range is {0, 1} 0 – do not snapshot on video channel <i>ch</i> 1 – snapshot on video channel <i>ch</i> |
| <i>handlerName</i> .EventHandler.SnapshotEnable | bool | Enable/Disable snapshot function. |
| <i>handlerName</i> .EventHandler.SnapshotPeriod | integer | Range is [0-255]. Frames between snapshot. 0 means continuously snapshot for every frame. |
| <i>handlerName</i> .EventHandler.SnapshotTimes | integer | Range is [0-65535] Snapshot times before stop, 0 means don't stop snapshot. |

| | | |
|--|--------|--|
| <i>handlerName</i> .EventHandler.TimeSection[<i>wd</i>][<i>ts</i>] | String | <p>It's table contains effective time period for eventHandler everyday. <i>wd</i> (week day) range is [0-6] (Sunday-Thursday) <i>ts</i> (time section) range is [0-23], it's index of time section table.</p> <p>Format: mask hh:mm:ss-hh:mm:ss Mask: {0,1}, hh: [0-24], mm: [00-59], ss: [00-59] Mask 0: this time section is not used. Mask 1: this time section is used.</p> <p>Example: TimeSection[1][0]=1 12:00:00-18:00:00 Means EventHandler is effective between 12:00:00 and 18:00:00 at Monday.</p> |
| <i>handlerName</i> .EventHandler.TipEnable | bool | Enable/Disable local message box tip. |

comment:

In above table, meaning of *handlerName* is the same with [6.1.2 Get Parameter\(EventHandler \)](#)

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&MotionDetect[0].Enable=true

6.1.2 Get Parameter

Get the parameter of EventHandler.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=<handlerName>
```

comment:

<handlerName> can be one of below four formats

Alarm[*alarm channel*].EventHandler
MotionDetect[*video channel*].EventHandler
BlindDetect[*video channel*].EventHandler
LossDetect[*video channel*].EventHandler

Example URL:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=Alarm[0].EventHandler
can get EventHandler settings of alarm channel 0.

Response:

```
handlerName.EventHandler.AlarmOut=1  
handlerName.EventHandler.AlarmOutChannels[0]=1  
handlerName.EventHandler.AlarmOutChannels[1]=1  
...  
handlerName.EventHandler.AlarmOutEnable=false  
handlerName.EventHandler.AlarmOutLatch=10  
handlerName.EventHandler.BeepEnable=true  
handlerName.EventHandler.Dejitter=0  
handlerName.EventHandler.Delay=30  
handlerName.EventHandler.LogEnable=true  
handlerName.EventHandler.MailEnable=true  
handlerName.EventHandler.PtzLink[0][0]=None  
handlerName.EventHandler.PtzLink[0][1]=0  
handlerName.EventHandler.PtzLink[1][0]=None  
handlerName.EventHandler.PtzLink[1][1]=0  
...  
handlerName.EventHandler.PtzLinkEnable=false  
handlerName.EventHandler.Record=1  
handlerName.EventHandler.RecordChannels[0]=1  
handlerName.EventHandler.RecordChannels[1]=1  
...  
handlerName.EventHandler.RecordEnable=true  
handlerName.EventHandler.RecordLatch=10  
handlerName.EventHandler.Snapshot=1  
handlerName.EventHandler.SnapshotChannels[0]=1  
handlerName.EventHandler.SnapshotChannels[1]=1  
...
```

```

handlerName.EventHandler.SnapshotEnable=false
handlerName.EventHandler.SnapshotPeriod=3
handlerName.EventHandler.SnapshotTimes=0
handlerName.EventHandler.TimeSection[0][0]=1 01:00:00-24:00:00
handlerName.EventHandler.TimeSection[0][1]=1 01:00:00-24:00:00
...
...
handlerName.EventHandler.TimeSection[6][5]=1 01:00:00-24:00:00
handlerName.EventHandler.TipEnable=true
    
```

Example:

```
/cgi-bin/paramManager.cgi?action=getParam&name=MotionDetect[0].EventHandler
```

6.2 Alarm

6.2.1 Set Alarm Parameter

Set parameter of alarm.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|---|-----------------|--|
| Alarm[<i>input</i>].Enable | bool | Enable/Disable alarm from a input channel |
| Alarm[<i>input</i>].EventHandler | | Setting of EventHandler is described in 6.1.1 Set Parameter (EventHandler) |
| Alarm[<i>input</i>].Name | string | Name of alarm input channel. |
| Alarm[<i>input</i>].SensorType | string | Range is {NC, NO}. NC: normal close NO: normal open |

comment:

In above table,
input is external alarm input channel,
ch is channel number,
wd is weekday index

ts is timesection index.

EventHandler defines parameter of relevant actions when alarm or event happens.

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&Alarm[0].EventHandler.AlarmOutEnable=true

6.2.2 Get Alarm Parameter

Get parameter of alarm.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=Alarm

Response:

```
table.Alarm[0].Enable=false
table.Alarm[0].EventHandler....(output of EventHandler is described in 6.1.2 Get Parameter \(EventHandler\))
table.Alarm[0].Name=Door1
table.Alarm[0].SensorType=NC
table.Alarm[1]....
...
```

6.2.3 Set Alarm Out Parameter

Set alarm out parameter in parameter tree.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]

With the following parameter and value type:

| paramName | paramValue type | Description |
|-----------|-----------------|-------------|
|-----------|-----------------|-------------|

| | | |
|------------------------------|---------|--|
| AlarmOut[<i>port</i>].Mode | Integer | Range is {0, 1, 2} 0: automatically alarm 1: force alarm 2: close alarm |
| AlarmOut[<i>port</i>].Name | String | Alarm out port name. |

comment:

Port in above table is alarm out port index, start form 0.

Response:

OK or ERROR

6.2.4 Get Alarm Out Parameter

Get the Alarm out parameter in parameter tree.

Method: GET

Syntax:

http://<*ip*>/cgi-bin/paramManager.cgi?action=getParam&name=**AlarmOut**

Response:

table.AlarmOut[*alarmOutChannel*].Mode=0
table.AlarmOut[*alarmOutChannel*].Name=Beep

comment:

*alarmOutChannel*above is the alarm out channel index.

6.2.5 Get In Slots

Get the solts of alarm input.

Method: GET

Syntax:

http://<*ip*>/cgi-bin/alarm.cgi?action=**getInSlots**

Response:

result=2

comment:

Get alarm input channel number.
Above response means there are 2 alarm input channels.

6.2.6 Get Out Slots

Get the alarm Solts number.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/alarm.cgi?action=getOutSlots
```

Response:

```
result=1
```

comment:

Get alarm output channel number.

6.2.7 Get In State

Get alarm input state for all channels.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/alarm.cgi?action=getInState
```

comment:

Response:

```
result=3
```

comment:

Get alarm input state for all channels.
A bit in the response result indicates a channel alarm states, above result 3 means alarm channel 1

and channel 2 have alarm now.

6.2.8 Get Out State

Get alarm output state for all channels.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/alarm.cgi?action=getOutState
```

Response:

```
result=0
```

comment:

Get alarm output state for all channels.

A bit in the response result indicates a channel. 1 means alarm is present.

6.3 MotionDetect

6.3.1 Set Parameter

Get the parameter of motion detect event.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/paramManager.cgi?action=setParam<paramName>=<paramValue>[&paramName=<paramValue>...]
```

With the following parameter and value type:

| paramName | paramValue type | Description |
|--------------------------|-----------------|---|
| <i>head.Enable</i> | bool | Enable/Disable motion detect feature in a channel. |
| <i>head.EventHandler</i> | | Setting of EventHandler is described in 6.1.1Set Parameter (EventHandler) |
| <i>head.Level</i> | integer | Range is [1-6]. Sensitivity of motion detection. |

| | | |
|---------------------------------------|---------|--|
| | | 1: lowest sensitivity. 6: highest sensitivity. |
| <i>head</i> .Region[<i>LineNum</i>] | integer | Currently, region is divided into 18 lines and 22 blocks/line. A bit describes a block in the line. Bit = 1: motion in this block is monitored.. Example: MotionDetect[0].Region[0] = 4194303 (0x3FFFFFF):: motion in channel 0 line 0's 22 blocks is monitored. MotionDetect[0].Region[1] =0: motion in line 1's 22 blocks is not monitored. MotionDetect[0].Region[17] = 3: in the last line of channel 0, motion in the left two blocks is monitored. |

comment:

Channel: video channel index

LineNum

Index of region, region is divided into lines and each line has several blocks, a line is described by a 32 bit integer, a bit for a block..

0=Line 1

1=Line 2

...

...

Head=MotionDetect[*Channel*]

The italics above will be replaced by the above abbreviations.

Response:

OK or ERROR

Example:

```
/cgi-bin/paramManager.cgi?action=setParam&MotionDetect[0].EventHandler.AlarmOutChannels[0]=0
```

6.3.2 Get Parameter

Get the parameter of motion detect events.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=**MotionDetect**

Response:

```
table.MotionDetect[0].Enable=false
table.MotionDetect[0].EventHandler... (output of EventHandler is described in 6.1.2 Get Parameter \(EventHandler\))
table.MotionDetect[0].Level=3
table.MotionDetect[0].Region[0]=4194303
table.MotionDetect[0].Region[1]=4194303
...
...
table.MotionDetect[1]...
...
```

comment:

MotionDetectparamof a video channel contains Enable, Level, Region and EventHandler.

6.4 Blind Detect

6.4.1 Set Parameter

Set the parameter of blind detect.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=setParam&<paramName>=<paramValue>[&<paramName>=<paramValue>...]

With the following parameter and value type:

| paramName | paramValue type | Description |
|---------------------------|-----------------|---|
| <i>head</i> .Enable | bool | Enable/Disable blind detect feature. |
| <i>head</i> .EventHandler | | Setting of EventHandler is described in 6.1.1Set Parameter (EventHandler) |
| <i>head</i> .Level | integer | Range is [1-6]. Sensitivity of blind detection. 1: lowest sensitivity. 6: highest sensitivity. |

comment:

Channel: video channel number

head=BlindDetect[*Channel*]

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&BlindDetect[0].Enable=true

6.4.2 Get Parameter

Get the information of blind detect events.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=**BlindDetect**

Response:

head.Enable=false

head.EventHandler= (output of EventHandler is described in [6.1.2 Get Parameter \(EventHandler\)](#))

head.Level=3

comment:

Channel: video channel number

head=table.BlindDetect[*Channel*]

6.5 Loss Detect

6.5.1 Set Parameter

Set the parameter of loss detect event.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=setParam<paramName>=<paramValue>[&<paramName>=<paramValue>...]

With the following parameter and value type:

| paramName | paramValue type | Description |
|---------------------------|-----------------|--|
| <i>head</i> .Enable | bool | Enable/Disable loss detect feature. |
| <i>head</i> .EventHandler | | Setting of EventHandler is described in 6.1.1 Set Parameter (EventHandler) |

comment:

Channel: video channel number
Head=LossDetect[*Channel*]

Response:

OK or ERROR

Example:

/cgi-bin/paramManager.cgi?action=setParam&LossDetect[0].Enable=true

6.5.2 Get Parameter

Get the parameter of loss detect.

Method: GET

Syntax:

http://<ip>/cgi-bin/paramManager.cgi?action=getParam&name=LossDetect

Response:

head.Enable=false
head.EventHandler= (output of EventHandler is described in [6.1.2 Get Parameter \(EventHandler\)](#))

comment:

Channel: video channel number
head=table.LossDetect[*Channel*]

6.6 Get Events Indexes

Get indexes of events.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/eventManager.cgi?action=getEventIndexes&code=<eventCode>
```

comment:

Get channels indexes that event of code eventCode happens.

eventCode includes:

VideoMotion: motion detection event

VideoLoss: video loss detection event

VideoBlind: video blind detection event.

Response:

```
channels[0]=0
```

```
channels[1]=2
```

```
channels[2]=3
```

```
...
```

(This response means event happened on channel 0, channel 2, and channel 3.)

7 System Operation

7.1 Reboot

Reboot the device.

Method: GET

Syntax:

```
http://<ip>/cgi-bin/system.cgi?action=reboot
```

comment:

Reboot the device. If successful, response OK. If fail, response ERROR.

Response:

OK or ERROR

Example:

/cgi-bin/ system.cgi?action=**reboot**

7.2 GetDeviceType

Get the device type.

Method: GET

Syntax:

http://<ip>/cgi-bin/ system.cgi?action=**getDeviceType**

Response:

OK or ERROR

If OK then type=IPC, continue to add other values

comment:

Get the device type.