

**PiXORD**

**H.264 Series**

**CGI Reference Manual**

**Version: 1.8**

**Date: 01/14/2011**

# TABLE OF CONTENTS

<b>TABLE OF CONTENTS .....</b>	<b>2</b>
<b>REVISION HISTORY .....</b>	<b>4</b>
<b>1 CGI COMMANDS .....</b>	<b>6</b>
1.1 PARAM.CGI .....	6
<i>Add, update, remove and list parameters (Describe how to get Mac/Brand/Model name).....</i>	<i>6</i>
1.2 PWDGRP.CGI .....	11
<i>Add, modify and delete users.....</i>	<i>11</i>
1.3 PTZ.CGI.....	13
<i>PTZ commands.....</i>	<i>13</i>
1.4 FILE.CGI.....	15
1.5 BACKUP_CONFIG.CGI.....	18
1.6 RESTART.CGI.....	18
1.7 FACTORYDEFAULT.CGI .....	18
1.8 SYSTEMLOG.CGI.....	18
1.9 SERVERREPORT.CGI.....	19
1.10 WIRELESS_SETTING.CGI.....	19
1.11 WIRELESS_CONNECT.CGI.....	19
1.12 SNAPSHOT.CGI .....	20
1.13 TRIGGER.CGI .....	20
1.14 AUDIO .....	20
1.15 MOTION .....	21
1.16 GPIOSTATUS.CGI .....	23
1.17 FE.CGI.....	24
<b>2 PARAMETER GROUPS.....</b>	<b>32</b>
2.1 IMAGESOURCE .....	32
2.2 IMAGE .....	34
2.3 OSD.....	36
2.4 AUDIO.....	37
2.5 AUDIOSOURCE.....	37
2.6 EVENTSERVERS.....	38
<i>FTP Server.....</i>	<i>38</i>
2.6.2 <i>HTTP Server.....</i>	<i>38</i>
2.6.3 <i>TCP Server.....</i>	<i>39</i>
2.7 EVENT CONFIGURATION.....	40
2.8 MOTION DETECTION .....	47

2.9	INPUT PORT .....	48
2.10	OUTPUT PORT .....	49
2.11	NETWORK GENERAL .....	49
2.12	NETWORK ADVANCE .....	50
2.13	SMTP(E-MAIL) .....	51
2.14	DDNS .....	53
2.15	PPPoE .....	54
2.16	BONJOUR .....	54
2.17	UPnP .....	55
2.18	HTTPS .....	56
2.19	WIRELESS .....	56
2.20	ANONYMOUS LOGIN .....	58
2.21	DATE & TIME .....	58
2.22	LANGUAGE .....	62
2.23	LOG CONFIGURATION .....	62
2.24	PTZ .....	63
2.25	LAYOUT .....	64
2.26	EXTERNAL VIDEO .....	67
<b>3</b>	<b>SERVER SIDE INCLUDE COMMANDS .....</b>	<b>70</b>
3.1	PINIT .....	70
3.2	PGET .....	70
3.3	PGET_GPIO .....	71
3.4	PGET_FORM .....	71
	<b>APPENDIX .....</b>	<b>73</b>
A.	RTSP URL .....	73
B.	STREAMS DEPENDENCY .....	75
C.	HTTP STATUS CODES .....	77

---

---

## *Revision History*

<b>Version</b>	<b>Release Date</b>	<b>Description</b>
<b>1.0</b>	<b>May.14 2009</b>	<b>Initial version</b>
<b>1.1</b>	<b>Jun.08.2009</b>	<b>Rename some CGIs and Params. Add stream descriptions.</b>
<b>1.2</b>	<b>Jun.18.2009</b>	<b>Streams Dependency Revision</b>
	<b>Jun.26.2009</b>	<b>Add “listgroup” action of param.cgi</b>
<b>1.3</b>	<b>Aug.06.2009</b>	<b>Add “motioninfo” CGI command</b>
		<b>Add “HTTPS” parameter group</b>
		<b>Add “Bonjour”, “UPnP” in Network parameter group.</b>
	<b>Sep.15.2009</b>	<b>Add Pget_gpio SSI</b>
		<b>Change the status of Input/Output parameter to “high/low” instead of “open/close”</b>
	<b>Oct.22.2009</b>	<b>Add “ICR” and “ExternalOutputFormat” in ImageSource group.</b>
<b>1.4</b>	<b>Dec.07.2009</b>	<b>Revise the Audio methods</b>
		<b>Modify the Event configuration of TCP</b>
<b>1.5</b>	<b>Apr.01.2010</b>	<b>Add how to get Mac address, brand name, model name of product in section</b>

<b>1.6</b>	<b>Jun.29.2010</b>	<b>1.1.1</b> <b>Add how to get input and output status (section 1.16), set output status (section 2.10) by new firmware, TCP payload message of event (section 2.7)</b>
<b>1.7</b>	<b>Nov.1.2010</b>	<b>Add fe.cgi (section 1.17) which controls the fish eye function.</b>
<b>1.8</b>	<b>Jan.13.2011</b>	<b>Add query and preset point function to fe.cgi. (section 1.17)</b>

This document is intended as a guide for application developers and describes how to use scripting in CGI Command of Network Camera or Server. The information is provided “as is” without warranty of any kind and is subject to change without notice. And also reserves the right to revise the content of this document at any time without prior notice.

---

---

# 1 CGI Commands

This CGI Command document specifies the method of communication with the Network Camera or Server for controlling camera functions as well as for getting and setting internal parameter values, which helps the application integrators develop software applications more easily.

## 1.1 param.cgi

*Add, update, remove and list parameters (Describe how to get*

*Mac/Brand/Model name)*

**Description:** Create or remove applicable for dynamic parameter groups such as the event parameters, modify the parameters value and list all of the parameters and their values.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/param.cgi?[parameter]=<value>[&primary group name[.primary group value][.secondary group name][.secondary group value].parameter name=<value>]
```

Parameter	Value	Description
actions	add, update , remove , list, listgroup	Specifies the action to take. Depending on this parameter, various parameters may be set, as described in the following sections.

		<p><b>Add</b> = create the new parameter.</p> <p><b>Update</b> = change the parameter value.</p> <p><b>Remove</b> = delete the parameter.</p> <p><b>List</b> = list the value of specified parameter or all the values if no any specified parameter.</p> <p><b>Listgroup</b>= list the values of specified parameter group, e.g. "Network".</p>
Group name	<String>	Specifies the group name
group value	<String>	Specifies the group value
parameter name	<String>	Specifies the parameter name.

**Example:**

E- **Add the parameter:**

**Method:** GET

<http://192.168.0.200/cgi-bin/admin/param.cgi?actions=add&EventServers.TCP.T00.Address=192.168.0.200&EventServers.TCP.T00.Name=NewTCPServer&EventServers.TCP.T00.Port=8080>

(2) **Update the parameter**

**Method:** GET

http://192.168.0.200/cgi-bin/admin/param.cgi?actions=update&EventServers.TCP.T00.Address=192.168.0.200&EventServers.TCP.T00.Name=NewTCPServer&EventServers.TCP.T00.Port=8080

(3) **Remove a parameter:**

**Method:** GET

```
http://192.168.0.200/cgi-bin/admin/param.cgi?actions=remove&EventServers.TCP.T00
```

**(4) List a parameter:**

**Method:** GET

```
http://192.168.0.200/cgi-bin/admin/param.cgi?actions=list&Network.IPAddress
```

**Return :**

```
Network.IPAddress= 192.168.1.83
```

Other example

**Method:** GET

```
http://192.168.0.200/cgi-bin/admin/param.cgi?actions=list&Network.MACAddress
```

**Method:** GET

```
http://192.168.0.200/cgi-bin/admin/param.cgi?actions=list&Brand.Brand
```

**Method:** GET

```
http://192.168.0.200/cgi-bin/admin/param.cgi?actions=list&Brand.ProdModel
```

**(5) List all parameter:**

**Method:** GET

```
http://192.168.0.200/cgi-bin/admin/param.cgi?actions=list
```

**Return:**

```
Audio.A00.AlarmLevel=2  
Audio.A00.Enabled=yes  
Audio.NbrOfAudios=1  
AudioSource.A00.BitRate=64000
```

```
AudioSource.A00.Encoder=g711u
AudioSource.A00.InputGain=26
AudioSource.A00.InputPreGain=32
AudioSource.A00.InputType=
AudioSource.A00.SampleRate=16000
Brand.Brand=
Brand.ProdFullName=
Brand.ProdNbr=P600DN
Brand.ProdShortName=
Brand.ProdType=
Brand.WebURL=
    :
    :
    :
Wireless.SubnetMask=
Wireless.WEPEncryption=64
Wireless.WPAEncryption=
Wireless.WPAKey=
```

**(6) List a group of parameters:**

**Method:** GET

<http://192.168.0.200/cgi-bin/admin/param.cgi?actions=listgroup&Motion>

**Return:**

```
Motion.M00.Bottom=159
Motion.M00.Color=FF0000
Motion.M00.History=0
Motion.M00.ImageSource=0
Motion.M00.Left=57
Motion.M00.Name=M1
Motion.M00.Right=201
Motion.M00.Sensitivity=0
Motion.M00.Top=40
Motion.M00.Trigger=0
Motion.M00.WindowType=include
Motion.M01.Bottom=199
```

Motion.M01.Color=0938F7  
Motion.M01.History=0  
Motion.M01.ImageSource=0  
Motion.M01.Left=233  
Motion.M01.Name=M2  
Motion.M01.Right=329  
Motion.M01.Sensitivity=0  
Motion.M01.Top=120  
Motion.M01.Trigger=0  
Motion.M01.WindowType=include  
Motion.NbrOfMotions=0,1

## 1.2pwdgrp.cgi

### *Add, modify and delete users*

**Description:** Add a new user with password and group membership, modify the information and remove a user.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/pwdgrp.cgi?<parameter>=<value>[&<parameter>=<value>
>...]
```

Parameter	Value	Description
actions	add, update , remove , list	<b>add</b> = create a new user account. <b>Update</b> = change account information of specified parameters if the account exists. <b>Remove</b> = remove an existing account if it <b>list</b> = list of the users which belong to each group defined.
User	<String>	The user account name.
pwd	<String>	The unencrypted password of the account.
Grp	pxadmin, pxoperator, pxview	An existing primary group name of the account. <b>Pxadmin</b> = Administrator. <b>Pxoperator</b> = Operator. <b>Pxview</b> = View.

*Example:*

**E- Create a new administrator account:**

```
http://192.168.0.200/cgi-bin/admin/pwdgrp.cgi?actions=add&user=test&pwd=test&grp=pxadmin
```

**(2) Modify the password of the administrator account.**

```
http://192.168.0.200/cgi-bin/admin/pwdgrp.cgi?actions=update&user=test&pwd=t1&grp=pxadmin
```

**(3) Delete a user:**

```
http://192.168.0.200/cgi-bin/admin/pwdgrp.cgi?actions=remove&user=test
```

**(4) List all of the users:**

```
http://192.168.0.200/cgi-bin/admin/pwdgrp.cgi?actions=list
```

***Return List :***

```
root=  
pxadmin=admin  
pxoperator=user1,user2  
pxview=user1,user2
```

## 1.3 ptz.cgi

### PTZ commands

**Description:** To control the Pan, Tilt and Zoom behavior of a PTZ unit, the following PTZ control URL is used.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/ptz.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

Parameter	Value	Description
action	ptz_play	Specifies the action to take.
Ch	0	The number of channel. e.g. Channel 1: ch=0
cmd	up, left, right, down, zoomin, zoomout, irisopen, irisclose, focusin, focusout, stop, autopan_start, autopan_stop, go_preset, save_preset, delete_preset	PTZ commands: <b>up</b> = Tilt-up move command. <b>Left</b> = Pan-left move command. <b>Right</b> = Pan-right move command. <b>Down</b> = Tilt-down move command. <b>Zoomin</b> = Do Zoom in command. <b>Zoomout</b> = Do Zoom out command. <b>Irisopen</b> = Do Iris open command. <b>Irisclose</b> = Do Iris close command. <b>Focusin</b> = Do Focus near command. <b>Focusout</b> = Do Focus far command. <b>Stop</b> = Stop command. <b>Autopan_start</b> = Start the Auto Pan / Zone scan command. <b>Autopan_stop</b> = Stop the Auto Pan / Zone scan command. <b>Go_preset</b> = Go to the preset point. <b>Save_preset</b> = Save the preset point. <b>Delete_preset</b> = Delete the preset point.
Point	0~19	The number of the PTZ preset position.

***Example:***

**(1) Stop the PTZ:**

[http://192.168.0.200/cgi-bin/admin/ptz.cgi?action=ptz\\_play&ch=0&cmd=stop](http://192.168.0.200/cgi-bin/admin/ptz.cgi?action=ptz_play&ch=0&cmd=stop)

**(2) Go to preset position 1:**

[http://192.168.0.200/cgi-bin/admin/ptz.cgi?action=ptz\\_play&ch=0&cmd=go\\_preset&point=1](http://192.168.0.200/cgi-bin/admin/ptz.cgi?action=ptz_play&ch=0&cmd=go_preset&point=1)

## 1.4file.cgi

**Description:** Upgrade the firmware version, upload a unit specific backup previously created by the **backup\_config.cgi**, or add/delete the language file.

**Method:** POST

**Enctype:** multipart/form-data

Parameter	Value	Description
actions	upgrade, restore, ddling, dellang	Specifies the action to take. <b>Upgrade</b> = Upgrade the firmware version. <b>Restore</b> = Upload a unit specific backup previously created by the backup_config.cgi. <b>ddling</b> = Add the language file. <b>Dellang</b> = Delete the language file
file	<type="file">	Base file name for uploaded files. e.g. <b>upgrade file:</b> flash_1_0_1_3441.bin <b>restore file:</b> config_backup.tgz <b>add language file:</b> lang.js
lang	ja, ko, th, ms, vi, id, hi, ru, sk, fr, it, de, pl, fi, nl, pt, da, no, lt, es, hr, iw, el, lv, ar, bg, uk, cs, sl, sr, sv, ro, tr, hu, is, la, ot	Select language to add. <b>Ja</b> = 日本語 <b>ko</b> = 한국어 <b>th</b> = ไทย <b>ms</b> = Bahasa Melayu <b>vi</b> = Tiếng Việt <b>id</b> = Bahasa Indonesia <b>hi</b> = □□□□□□ <b>ru</b> = Русский <b>sk</b> = Slovenčina <b>fr</b> = Français <b>it</b> = Italiano <b>de</b> = Deutsch <b>pl</b> = Polski <b>fi</b> = Suomi

		<b>nl</b> = Nederlands <b>pt</b> = Português <b>da</b> = Dansk <b>no</b> = Norsk(Bokmål) <b>lt</b> = Lietuvių <b>es</b> = Español <b>hr</b> = Hrvatski <b>iw</b> = עברית <b>el</b> = Ελληνικά <b>lv</b> = Latviešu <b>ar</b> = العربية <b>bg</b> = Български <b>uk</b> = Українська <b>cs</b> = Čeština <b>sl</b> = Slovenščina <b>sr</b> = Српски / Srpski <b>sv</b> = Svenska <b>ro</b> = Română <b>tr</b> = Türkçe <b>hu</b> = Magyar <b>is</b> = Íslenska <b>la</b> = Latina <b>ot</b> = Other
rurl	<URL>	Content source for return after uploading files. e.g., <a href="http://192.168.0.200/admin/update.html">http://192.168.0.200/admin/update.html</a>

**Example:**

**E- Upgrade firmware:**

```

<form name="upfwfm" method="POST" enctype="multipart/form-data"
action="/cgi-bin/admin/file.cgi">
  <input type="hidden" name="actions" value="upgrade">
  <input type="hidden" name="rurl" value="">
  <input type="file" name="file" >
  <input name="Submit" type="submit" value="Upgrade" >
</form>

```

## (2) Upload of backup:

```
<form name="uploadfm" method="POST" enctype="multipart/form-data"
action="/cgi-bin/admin/file.cgi">
  <input type="hidden" name="actions" value="restore">
  <input type="hidden" name="rurl" value="">
  <input type="file" name="file">
  <input type="submit" name="upgradebtn" value="Upgrade">
</form>
```

## (3) Upload of language:

```
<form name="uplangfm" method="POST" enctype="multipart/form-data"
action="/cgi-bin/admin/file.cgi">
  <input type="hidden" name="actions" value=" ddling">
  <input type="hidden" name="rurl" value="">
  <input type="hidden" name="lang" value="ja">
  <input type="file" name="file">
  <input type="submit" name="upgradebtn" value="Upload Language">
</form>
```

## (4) Remove language:

```
<form name="uplangfm" method="POST" enctype="multipart/form-data"
action="/cgi-bin/admin/file.cgi">
  <input type="hidden" name="actions" value="dellang">
  <input type="hidden" name="rurl" value="">
  <input type="hidden" name="lang" value="ja">
  <input type="file" name="file">
  <input type="submit" name="upgradebtn" value="Remove Language">
</form>
```

## ***1.5 backup\_config.cgi***

**Description:** Download a unit specific backup of all files in the folder /etc in tgz format.

**Method:** POST

**Example:**

```
<form name="form_export" method="post" action="/cgi-bin/admin/backup_config.cgi">
  <input name="Submit" type="submit" value="Backup" >
</form>
```

## ***1.6 restart.cgi***

**Description:** Restart server.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/restart.cgi
```

## ***1.7 factorydefault.cgi***

**Description:** Reload factory default.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/factorydefault.cgi
```

## ***1.8 systemlog.cgi***

**Description:** Get system log information.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/systemlog.cgi
```

## ***1.9 serverreport.cgi***

**Description:** This CGI request generates and returns a server report. This report is useful as an input when requesting support. The report includes product information, parameter settings and system logs.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/serverreport.cgi
```

## ***1.10 wireless\_setting.cgi***

**Description:** Get status of Wireless Network.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/wireless_setting.cgi
```

**Example:**

```
http://192.168.0.200/cgi-bin/admin/wireless_setting.cgi
```

## ***1.11 wireless\_connect.cgi***

**Description:** Connect the Wireless Network.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/wireless_connect.cgi
```

**Example:**

```
http://192.168.0.200/cgi-bin/admin/wireless_connect.cgi
```

## 1.12 *snapshot.cgi*

**Description:** Request a JPEG image (snapshot) with specified properties.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/snapshot.cgi
```

## 1.13 *trigger.cgi*

**Description:** Trigger events manually.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/trigger.cgi
```

**Example:**

```
http://192.168.0.200/cgi-bin/admin/trigger.cgi
```

## 1.14 *audio*

**Description:** Push audio from PC to IP Camera. Audio type: uLaw 8K

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/audio?<parameter>=<value>
```

**Example:**

**(1) Play voice if ipcam is stopping:**

```
http://<servername>/cgi-bin/admin/audio?KAUD=PLAY
```

**(2) Stop voice if ipcam is playing:**

```
http://<servername>/cgi-bin/admin/audio?KAUD=STOP
```

**(3) Force to stop voice:**

```
http://<servername>/cgi-bin/admin/audio?KAUD=RESET
```

## 1.15 Motion

**Description:** Retrieves the current motion detection level from all or specific motion detection including windows. A motion detection window is defined by a dynamic parameter group Motion.M<group number>.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/motioninfo
```

**Return:**

HTTP Code: 200 OK

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

Body:

```
--<boundary> <motion levels>
where the returned <motion levels> part is
Content-Type: text/plain <motion level for window with lowest group number> --<boundary>
and <motion level for window with group number n> is
group=<group number n>;level=<motion level for n>;threshold= <threshold level for n>;
[ <motion level for window n+1> ]
```

**Example:**

HTTP Code: 200 OK

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

Body:

```
Content-Type: text/plain

group=0;level=10;sensitivity=40;
group=1;level=40;sensitivity=80;

--motinfo

Content-Type: text/plain

group=0;level=10;sensitivity=40;
group=1;level=40;sensitivity=80;
```

--motinfo

Content-Type: text/plain

group=0;level=12;sensitivity=40;

group=1;level=46;sensitivity=80;

--motinfo

## 1.16 *gpiostatus.cgi*

**Description:** get GPIO status.

**Method:** GET

**Syntax:**

```
http://<servername>/cgi-bin/admin/gpiostatus.cgi
```

**Return:**

HTTP Code: 200 OK

```
Line-based text data: text/plain  
Input.I00.Status=high  
Input.I01.Status=high  
Output.O00.Status=low
```

**Example:**

```
http://192.168.0.200/cgi-bin/admin/gpiostatus.cgi
```

## 1.17 *fe.cgi*

**Description:** To configure Fish Eye functions.

**Method:** GET

### 1. Query the information of Fish Eye IP Camera.

Query function is used to query the information of the Fish Eye IP Camera.

#### A. Query the camera setup:

User can pass **camera\_info** command to query the display mode, mounting type and saved preset points information of IP Camera. If CGI executes successfully, camera will return the information of mounting type, display mode and available preset points. If preset point function is not supported in current display mode, preset points return -1.

Parameter	Value	Description	
action	query	query the information of Fish Eye	
cmd	camera_info	query all information of the IP Camera	
Return	Item	Value	Description
	MountType	ceiling, wall, table	current mounting type
	DisplayMode	quad, quad_source, triple, double, panorama, original	current display mode
	PresetPoints	saved preset points	All saved preset points, divided by comma, not necessary by order (ex: 0,5,3,10)

**Example:**

- **Query the camera setup:**

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=query&cmd=camera_info
```

**Return:**

HTTP Code: 200 OK

Line-based text data: text/plain

```
MountType=ceiling&DisplayMode=quad&PresetPoints=3,0,4,10,15
```

- **Query the camera setup (double mode doesn't support preset points):**

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=query&cmd=camera_info
```

**Return:**

HTTP Code: 200 OK

Line-based text data: text/plain

```
MountType=ceiling&DisplayMode=double&PresetPoints=-1
```

**B. Query the Pan, Tilt, and Zoom information of the IP Camera:**

User can pass **ptz\_info** command to query the PTZ information of IP Camera by channel and the value of Pan, Tilt and Zoom are returned. Note that in some modes, Tilt and Zoom are not supported and the returned value of Tilt and Zoom are -1. If the PTZ function is not supported in current channel and display mode, or user passes invalid ch parameter, camera won't return anything.

Parameter	Value	Description	
action	query	Query the information of Fish Eye	
cmd	ptz_info	Query PTZ information of IP Camera	
ch	number of channel	query the PTZ info of the channel, range from 0 to 3	
Return	Item	Value	Description
	ChX	<Pan>,<Tilt>,<Zoom>	Pan, Tilt, Zoom values divided by comma of specified channel

**Example:**

- **Query the PTZ information of channel 1:**

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=query&cmd=ptz_info&ch=1
```

**Return:**

HTTP Code: 200 OK

Line-based text data: text/plain

```
Ch1=180,300,5
```

- **Query the PTZ information of channel 2: (assume panorama mode, Tilt and Zoom function are not supported)**

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=query&cmd=ptz_info&ch=2
```

**Return:**

HTTP Code: 200 OK

Line-based text data: text/plain

Ch1=135,-1,-1

**C. Query the preset point information of the IP Camera:**

User can pass **preset\_info** command to query the preset point information of the IP Camera. By now, only quad and quad\_source modes in ceiling mount support preset point function.

Parameter	Value	Description	
action	query	Query the information of Fish Eye	
cmd	preset_info	Query preset point information of IP Camera	
point	preset point number	query the preset point info, range from 0 to 15	
Return	Item	Value	Description
	ChX~ChY	<Pan>,<Tilt>,<Zoom>	Pan, Tilt, Zoom values divided by comma of all channels The number of channels depends on the display mode. (ex: in quad view, 4 channels 0~3 are returned, while in quad_source view 3 channels 1~3 are returned)

**Example:**

● **Query the preset point information of preset point 5:**

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=query&cmd=preset_info&point=5
```

**Return:**

HTTP Code: 200 OK

Line-based text data: text/plain

Ch0=180,300,5&Ch1=90,200,3&Ch2=235,250,1&Ch3=230,224,2

## 2. Change the mounting type of fisheye IP Camera:

Each mounting type has a default display mode, the default display mode of **ceiling** mount is **quad**, of **wall** mount is **quad\_source**, and of **table** mount is **double**.

If CGI executes successfully, the current mount type, current display mode and saved preset points are returned. If preset point function is not supported or no preset point is saved in this display mode, -1 is returned. If user passes an invalid mount type, camera will return the display mode, mount type, and preset point information of current state.

Parameter	Value	Description
action	mount_type	Specify the Fish Eye mounting type.
cmd	ceiling, wall, table	Fish Eye IPcam mounting type: <b>ceiling</b> : mount on ceiling, default display mode is <b>quad</b> <b>wall</b> : mount on wall, default display mode is <b>quad_source</b> <b>table</b> : mount on table, default display mode is <b>double</b>

*Example:*

### ● Set Fish Eye IP Camera to **ceiling** mounting type (no preset points saved):

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=mount_type&cmd=ceiling
```

*Return:*

HTTP Code: 200 OK

```
Line-based text data: text/plain
```

```
DisplayMode=quad&MountType=ceiling&PresetPoints=-1
```

## 3. Change the display mode of IP Camera:

If CGI executes successfully, the current mount type and display mode are returned. If user assigns an invalid display mode, the returned mounting type and display mode is the current mounting type and display mode.

### A. Change the display mode of IP Camera in **ceiling** mount.

By now, only quad and quad\_source modes support preset point function. In other modes, PresetPoints always return -1.

Parameter	Value	Description
action	display_mode	Specify the Fish Eye display mode.

cmd	quad, quad_source, double, original, panorama,	Available display mode in <b>Ceiling</b> mount: <b>quad</b> = 360° quad view <b>quad_source</b> = 360° 3 PTZ view with source <b>double</b> = 360° double broad view <b>original</b> = source image <b>panorama</b> = 360° broad view
-----	---------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

*Example:*

- In **ceiling** mount, set IP Camera display mode to **quad**:

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=display_mode&cmd=quad
```

*Return:*

HTTP Code: 200 OK

Line-based text data: text/plain

```
DisplayMode=quad&MountType=ceiling&PresetPoints=2,3,5,7
```

### B. Change the display mode of IP Camera in **wall** mount.

By now, no mode in wall mount supports preset point function, so PresetPoints always return -1.

Parameter	Value	Description
action	display_mode	Specify the Fish Eye display mode.
cmd	quad_source, triple, original, panorama,	Available display mode in <b>Wall</b> mount: <b>quad_source</b> = 360° 3 ePTZ view with source <b>triple</b> = 180° broad view with two ePTZ view <b>original</b> = source image <b>panorama</b> = 360° broad view

*Example:*

- In **wall** mount, set IP Camera display mode to **quad\_source**:

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=display_mode&cmd=quad_source
```

*Return:*

HTTP Code: 200 OK

Line-based text data: text/plain

```
DisplayMode=quad_source&MountType=wall&PresetPoints=-1
```

### C. Change the display mode of IP Camera in Table mount.

By now, no mode in table mount supports preset point function, so PresetPoints always return -1.

Parameter	Value	Description
action	display_mode	Specify the Fish Eye display mode.
cmd	double, original	Available display mode in <b>Table</b> mount: <b>double</b> = double 360° broad view (video conference mode) <b>original</b> = source image

*Example:*

In **table** mount, set IP Camera display mode to **double**:

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=display_mode&cmd=double
```

*Return:*

HTTP Code: 200 OK

Line-based text data: text/plain

```
DisplayMode=double&MountType=table&PresetPoints=-1
```

#### 4. Digital Pan Tilt Zoom fisheye IP Camera

By now, PTZ only support **ceiling** mount, and support display modes are **quad**, **quad\_source**, **double**, and **panorama**. Note that **Pan** and **Tilt** are offset value, and **Zoom** is absolute value. If CGI executes successfully, the Pan, Tilt and Zoom (if available) will return. Note that in some display mode, zoom function is not available and will return -1.

##### A. Ceiling mount:

If CGI executes successfully, the Pan, Tilt and Zoom will return. The return format is ChX=Pan\_value,Tilt\_value,Zoom\_value, X is the channel number. The range of Pan\_value is 0~360, the range of Tilt\_value is 0~600 and Zoom\_value is 1~10. In some mode (ex: panorama), Tilt and Zoom function are no available, then the returned Tilt value and Zoom value are set to -1.

Parameter	Value	Description
action	eptz	Fish Eye PTZ function
cmd	move, reset,	<b>move:</b> eptz by pan, tilt, and zoom <b>reset:</b> reset the ptz setting of the given channel

ch	0, 1, 2, 3	Support channel in different display modes: <b>quad:</b> 0, 1, 2, 3 <b>quad_source:</b> 1, 2, 3 <b>double:</b> 0, 1 <b>panorama:</b> 0
pan	Pan offset	Support display modes: <b>quad, quad_source, double, panorama.</b> <b>+pan:</b> pan right <b>-pan:</b> pan left
tilt	Tilt offset	Support display modes: <b>quad, quad_source.</b> <b>+tilt:</b> tilt down <b>-tilt:</b> tilt up
zoom	Zoom ratio	Support display modes: <b>quad, quad_source</b> Zoom ratio from <b>1</b> (original) to <b>10</b> (max zoom in)

**Example:**

- **Pan, tilt and zoom the image in channel 0:**

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=eptz&cmd=move&ch=0&pan=10&tilt=-10&zoom=4
```

**Return:**

HTTP Code: 200 OK

```
Line-based text data: text/plain
Ch0=310,190,4
```

- **Reset the Pan, tilt and zoom value of channel 2:**

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=eptz&cmd=reset&ch=2
```

**Return:**

HTTP Code: 200 OK

```
Line-based text data: text/plain
Ch2=90,300,1
```

## 5. Save, Goto and Delete the ePTZ preset point of IP Camera

The ePTZ preset function provides a way to **save, change** and **delete** the ePTZ settings of **all channels** in a mode. If user passes invalid preset point or preset point function is not supported in current display mode, "PresetPoints=-1" is returned.

Parameter	Value	Description
action	eptz	Specify the Fish Eye display mode.
cmd	save_preset, go_preset, delete_preset	<b>save_preset:</b> save the current ePTZ setting (all channels) and return all saved preset points <b>go_preset:</b> change the ePTZ setting, and return PTZ settings of all channels, display mode and mounting type <b>delete_preset:</b> delete the ePTZ setting and return all saved preset points
point	0 ~ 15	We now support 16 preset points in a mounting type.

**Example:**

- **Save** the current PTZ setting to preset point 0 (we already saved points 2, 4, 5):

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=eptz&cmd=save_preset&point=0
```

**Return:**

HTTP Code: 200 OK

Line-based text data: text/plain

PresetPoints=0,2,4,5

- **Change** the current PTZ setting to those saved in preset point 0:

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=eptz&cmd=go_preset&point=0
```

**Return:**

HTTP Code: 200 OK

Line-based text data: text/plain

PresetPoints=0&Ch0=180,240,2&Ch1=90,200,1&Ch2=270,100,7&Ch3=45,150,1&DisplayMode=quad&MountType=ceiling

- **Delete** the current PTZ setting of preset point 15 (we already saved points 3, 5, 8, 15):

```
http://192.168.0.200/cgi-bin/admin/fe.cgi?action=eptz&cmd=delete_preset&point=15
```

**Return:**

HTTP Code: 200 OK

Line-based text data: text/plain

PresetPoints=3,5,8

## 2 Parameter Groups

### 2.1 ImageSource

**Description:** Parameters for each CCD/CMOS image source. This parameter group is product dependent and only available in network cameras. Check the product specification for supported parameters, default values and valid values.

**CGI:** param.cgi

**Primary group name:** ImageSource

**Primary group value:** I00 => Channel 1

[ImageSource. ] [ImageSource.I00. ]

Parameter	Value	Description
NbrOfSources	1	The number of Source.
Brightness	-100~100 (Default <b>0</b> )	The image brightness.
Contrast	-100~100 (Default <b>0</b> )	The image contrast.
HueAngle	-100~100 (Default <b>0</b> )	The image Hue Angle .
Saturation	-100~100 (Default <b>0</b> )	The image saturation.
WhiteBalance	auto, hold, fixed_outdoor1, fixed_outdoor2, fixed_indoor, fixed_fluor1, fixed_fluor2 (Default <b>auto</b> )	The image white balance.
Exposure	auto, flickerfree50, flickerfree60 (Default <b>auto</b> )	The image exposure.
MaxExposureTime	0~6 (Default <b>1</b> )	The maximum exposure time (1/60, 1/30, 1/15, 1/6, 1/2, 1,2 s)

MaxGain	0~4 (Default <b>0</b> )	Decides how much gain the exposure control algorithm is allowed to use.  ( 36, 31, 25, 19, 13 dB)
Mirror.Hori	yes, no	Enabled of mirror vertically.
Mirror.Vert	yes, no	Enabled of mirror horizontally.
WhiteLightLED	auto, alwayson, alwaysoff (Default <b>auto</b> )	White LED control.  ( for P605/605W/606/606W)
PowerLED	on, off (Default <b>on</b> )	Power indicator to turn on/off.  ( for P605/605W/606/606W)
StatusLED	on, off (Default <b>on</b> )	Link status indicator to turn on/off.  ( for P605/605W/606/606W)
ICR	auto, alwayson, alwaysoff (Default <b>auto</b> )	ICR control.  ( for P600DN)
Color	auto, alwayson, alwaysoff (Default <b>auto</b> )	Color mode control  ( for P600DN)
ExternalOutputFormat	pal, ntsc  (Default <b>pal</b> )	Video output format, <b>NTSC / PAL</b>  ( for P600DN)

## 2.2 Image

**Description:** Common image parameters used for all image configurations.

**CGI:** param.cgi

**Primary group name:** Image

**Primary group value:** I00~I02 => Stream ID (Stream 1 ~ Stream 3)

**Secondary group value:** G00 => Image Source

[Image.I00.G00. ]

Parameter	Value	Description
ImageSource	0~3	Video channel Source. Must be the same as the Group value.
Appearance.Resolution	1600x1200, 1280x720, 800x600, 720x576, 720x480, 640x480, 352x240, 320x240	The image resolution. ( 1600x1200, 1280x720 and 800x600 only for camera )
Enabled	yes, no	Stream Enabled or not.
Stream.Format	h264, mjpeg	Video Compression Format H264 & MJPEG.
GOV.Length	1~150	The length of the Group of Pictures.
GOV.Structure	I,IP	The structure of the Group of Pictures.  <b>I</b> = I-frame only  <b>IP</b> = I-frame and P-frame
RateControl.Mode	vbr, cbr	Specifies whether the rate controller operates in Variable Bit Rate (VBR) or constant bit rate (CBR) mode.  <b>Vbr</b> = Variable Bit Rate  <b>cbr</b> = Constant Bit Rate

Appearance.Compression	best, good, standard	Quality Level, for MJPEG or VBR mode.
Stream.FPS	5~30	Frames per second.  (1600x1200 : 5~15,  others :5~30 )
Stream.Priority	quality, efficiency	Whether the rate controller should prioritize quality or efficiency.
RateControl.TargetBitrate	64~10000	The target bit rate, in kbit/s, for CBR mode.
Stream.SessionPath	<String>	The URI path name for each stream. (Default: <b>stream 1:</b> v00 <b>stream 2:</b> v01 <b>stream 3:</b> v02 )

## 2.3 OSD

**Description:** Image text overlay parameters for each image configuration.

**CGI:** param.cgi

**Primary group name:** Image

**Primary group value:** I00 => Steam ID

**Secondary group name:** Text

[Image.I00.Text. ]

Parameter	Value	Description
Enabled	yes, no	OSD Enabled or not.
TextEnabled	yes, no	Enable inclusion of camera name in th image.
String	<String>	The text to show at the Position in the image.
DateEnabled	yes, no	Enabled inclusion of Date/Time in th image.

## 2.4 Audio

**Description:** Audio configuration. The name of the audio configuration, which audio source it is connected to and whether it should be enabled or not.

**CGI:** param.cgi

**Primary group name:** Audio

**Primary group value:** A00

[Audio. ] [Audio.A00. ]

Parameter	Value	Description
NbrOfAudios	1	The number of Audios.
Enabled	yes, no	Enable inclusion of camera name in th image. Enable/disable this audio configuration.

## 2.5 AudioSource

**Description:** Parameters for each audio source.

**CGI:** param.cgi

**Primary group name:** AudioSource

**Primary group value:** A00

[AudioSource.A00. ]

Parameter	Value	Description
InputGain	mute, 0, 6, 10, 14, 17, 20, 26, 32	Gain settings in dB for sound received from client.
BitRate	32000, 64000, 96000	The output bit rate (bits per second) from the encoder.
Encoder	aac, g711a, g711u	The audio codec to use. (AAC, G711-A-LAW, G711-Mu-Law)

## 2.6 EventServers

### FTP Server

**Description:** This group defines an FTP server that can be used by an event to upload files to.

**CGI:** param.cgi

**Primary group name:** EventServers

**Secondary group name:** FTP

**Secondary group value:** F00~ F

[EventServers.FTP. ] [EventServers.FTP.F00. ]

Parameter	Value	Description
NbrOfServers	0~9	The number of FTP server.
Name	<String>	A descriptive FTP server name.
Address	An IP address or a host name.	FTP server address.
Login	<String>	FTP user name.
Password	<String>	FTP password.
UploadPath	<String>	Directory where uploaded files go.
Port	1~65535	FTP server port.
Passive	yes, no	Use passive FTP.

### 2.6.2 HTTP Server

**Description:** This group defines an HTTP server that can be used by an event to upload files to

or send notification messages to.

**CGI:** param.cgi

**Primary group name:** EventServers

**Secondary group name:** HTTP

**Secondary group value:** H00~

[EventServers.HTTP. ] [EventServers.HTTP.H00. ]

Parameter	Value	Description
NbrOfServers	0~9	The number of HTTP server.
Name	<String>	A descriptive HTTP server name.
Address	<URL>	URL to the server, including name of CGI script to handle the request. (http://.....)
Login	<String>	HTTP user name.
Password	<String>	HTTP password.
Proxy	<IP address>	Proxy address.
ProxyPort	0~65535	Proxy port number.
ProxyLogin	<String>	Proxy user name.
ProxyPassword	<String>	Proxy password.

### 2.6.3 TCP Server

**Description:** This group defines a server that can be used by an event to send TCP/IP notification messages to.

**CGI:** param.cgi

**Primary group name:** EventServers

**Secondary group name:** TCP

**Secondary group value:** T00~

[EventServers.TCP. ] [EventServers.TCP.T00. ]

Parameter	Value	Description
NbrOfServers	0~9	The number of TCP server.
Name	<String>	A descriptive TCP server name.
Address	<IP address or host name>	IP address or host name of the TCP server.
Port	0~65535	TCP port number to use.

## 2.7 Event Configuration

**Description:** This group defines an event, which is a set of parameters describing how and when the product performs certain actions.

**CGI:** param.cgi

**Primary group name:** Event

**Primary group value:** E00~

**Secondary group name:** Actions

**Secondary group value:** A00~ A04

[Event. ] [Event.E00. ]

Parameter	Value	Description
NbrOfEvents	0~9	The number of Event.
IncludePreTrigger	yes, no	Use pre-trigger buffer.
PreTriggerSize	0~	Size of the pre-trigger buffer (in number of frames).
IncludePostTrigger	yes, no	Use post-trigger buffer.
PostTriggerSize	0~	Size of the post-trigger buffer (in number of frames).

Type	S, T	Scheduled or Triggered event.
Name	<String>	A descriptive event name.
Priority	0, 1, 2	Execution priority:  <b>0</b> = low. <b>1</b> = normal. <b>2</b> = high.
Active	yes, no	If the event is currently triggered.
MinimumTriggerInterval	00:00:00~ 23:59:59	Time interval between triggers in format “hours:minutes:seconds”, any triggers that occur during the interval are ignored. Max interval is 23:59:59.
Enabled	yes, no	Event enabled (disabled events are never triggered).
Weekdays	0000000,1000 000,0100000,  0010000,0001 000,0000100, 0000010,0000 001	Pattern of weekdays when the event should trigger.  (Sun.Mon.Tue.Wed.Thu.Fri.Sat )
Starttime	00:00:00~ 23:59:59	Time when the event should start to trigger in format “hh:mm:ss”.Max time is 23:59:59.
Duration	000:00:00~ 168:00:00	How long the event should trigger in format “hours:minute:second”. Max 168:00:00.
HWInputs	xxxx, 0xxx, x1xx, xx2x, xxx3	Hardware input trigger pattern.  <b>X</b> = do not trigger. <b>0</b> = trigger on inactivation. <b>1</b> = trigger on activation. <b>2</b> = trigger on activation change.

SWInput	IO5:/, M1:/, BOOT:/, log  (Trigger by: Input ports: "" Manual trigger:IO5/ Motion Detection:M1: / M1:\ M1:x  On Boot:BOOT:/ Log:log)	Software input trigger pattern.  <b>Virtual Input (manual):</b> IO<no>:<trigger> where the number can be an integer starting from 5, and the trigger can be “/” or “H” for activated and “\” or “L” for inactivated. <b>Example 1:</b> “IO5:H” means trigger when virtual input 5 is active. <b>Example 2:</b> “IO6:L” means trigger when virtual input 6 is inactive.  <b>BOOT:/</b> means trigger on boot. <b>Video loss:</b> V<channel no>:\ where channel number can be: 0 ... 3 = trigger when video is lost on the numbered video channel (counted from 0) <b>x</b> = trigger when video is lost on any video channel. <b>Example 1:</b> “V0:\” means trigger when video is lost on video channel 1. <b>Example 2:</b> “Vx:\” means trigger when video is lost on any video channel.  <b>Motion detection:</b> M<motion window no>:<trigger> where trigger can be: / = when motion starts. \ = when motion stops. <b>x</b> = when motion starts or stops. <b>Example 1:</b> “M1:/” means trigger when motion is detected for motion window 1. <b>Example 2:</b> “M1:x” means trigger when motion starts or stops for motion window 1.
UploadFormat	jpg, avi	Format for uploaded files.
FileName	<String>	Base file name for uploaded files.

A00.Type	U	Type of action. U = Upload.
A00.Protocol	FTP ,HTTP , SMTP	Protocol.
A00.Order	0~2	Execution order. If this event has multiple actions, this parameter determines when this action should be executed. Actions with Order=0 will be executed in parallel (asynchronously). Other actions will be executed synchronously, i.e. one action must finish before next action is executed, in rising order, i.e. first Order=1, then Order=2.
A00.Server	blank, F0 ... Fn (n = number of FTP event servers – 1)	Primary FTP server ID. Refers to a parameter group under root.EventServers.FTP. Example: “F0” refers to the parameter group root.EventServers.FTP.F0.
A00.Server2	blank, F0 ... Fn (n = number of FTP event servers – 1)	Secondary FTP server ID.
A00.EmailTo	<E-mail address>	E-mail address.
A00.Message	<String>	Message to send.
A00.ImagesPerMail	0~	Max number of images to upload in one e-mail.
A01.Type	N	Type of action.  N = Notification.
A01.Protocol	HW	Protocol.

A01.Order	0~2	Execution order. If this event has multiple actions, this parameter determines when this action should be executed. Actions with Order=0 will be executed in parallel (asynchronously). Other actions will be executed synchronously, i.e. one action must finish before next action is executed, in rising order, i.e. first Order=1, then Order=2.
A01.Output1	0, 1	Output1 to activate.  <b>0:Disabled</b>  <b>1:Enabled</b>
A01.KeepActive1	0, 1	Keep output 1 to active.  <b>0:Disabled</b>  <b>1:Enabled</b>
A01.Duration1	1~	How long the output should be active in seconds.
A02.Type	N, U	Type of action.  <b>N</b> = Notification. <b>U</b> = Upload.
A02.Protocol	SMTP	Protocol.
A02.Order	0~2	Execution order. If this event has multiple actions, this parameter determines when this action should be executed. Actions with Order=0 will be executed in parallel (asynchronously). Other actions will be executed synchronously, i.e. one action must finish before next action is executed, in rising order, i.e. first Order=1, then Order=2.
A02.EmailTo	<E-mail address.>	E-mail address.
A02.Subject	<String>	Subject line.

		<p>Possible to use dynamic event notification:</p> <p><b>#e</b> = The event name.  <b>#E</b> = The event description. Specified in the file /etc/event_desc.list located in the product's file system.  <b>#w</b> = The motion detection window.  <b>#t</b> = The trigger type and value.  Also other format codes such as file naming and date/time formats can be used.</p>
A02.Message	<String>	<p>Message body.</p> <p>Possible to use dynamic event notification:</p> <p><b>#e</b> = The event name.  <b>#E</b> = The event description. Specified in the file /etc/event_desc.list located in the product's file system.  <b>#w</b> = The motion detection window.  <b>#t</b> = The trigger type and value.  Also other format codes such as file naming and date/time formats can be used.</p>
A03.Type	N, U	<p>Type of action.</p> <p><b>N</b> = Notification.  <b>U</b> = Upload.</p>
A03.Protocol	HTTP	<p>Protocol.</p>
A03.Order	0~2	<p>Execution order. If this event has multiple actions, this parameter determines when this action should be executed. Actions with Order=0 will be executed in parallel (asynchronously). Other actions will be executed synchronously, i.e. one action must finish before next action is executed, in rising order, i.e. first Order=1, then Order=2.</p>

A03.Server	H0 ... Hn (n = number of HTTP event servers -1)	HTTP server ID. Refers to a parameter group under root.EventServers.HTTP. Example: "H0" refers to the parameter group root.EventServers.HTTP.H0.
A03.CustomParams	<String>	Custom parameters to add to URL. Example: "foo=bar". Spaces are not allowed in this field and all text must be URI-encoded ( <a href="#">RFC2396</a> ). Example: to set the CGI parameter 'example' to 'Y & Z' enter example=Y+%26+Z in this field.
A04.Type	N	Type of action.  N = Notification.
A04.Protocol	TCP	Protocol.
A04.Order	0~2	Execution order. If this event has multiple actions, this parameter determines when this action should be executed. Actions with Order=0 will be executed in parallel (asynchronously). Other actions will be executed synchronously, i.e. one action must finish before next action is executed, in rising order, i.e. first Order=1, then Order=2.
A04.Server	<IP address or host name>	IP address or host name of the TCP server.
A04.Port	0~65535	TCP port number to use.
A04.Count	0~4	Connect retry times.
A04.Message	TCP payload message	TCP payload message

## 2.8 Motion Detection

**Description:** Contain dynamic parameters, which define the properties of a Motion Detection window.

**CGI:** param.cgi

**Primary group name:** Motion

**Primary group value:** M00~

[Motion.] [Motion.M00. ]

Parameter	Value	Description
NbrOfMotions	0~9	The number of Motion Detection.
Name	<String>	A user's window identification, e.g. "Door" or "Window".
Left	0~351	The coordinate for the left boundary of the rectangular motion detection window. The full value range is related to the full image width and 0 is counted from the left hand side of the image.
Right	1~352	The coordinate for the right boundary of the rectangular motion detection window. The full value range is related to the full image width and 1 is counted from the left hand side of the image.
Top	0~239	The coordinate for the upper boundary of the rectangular motion detection window. The full value range is related to the full image height and 0 is counted from the upper side of the image.
Bottom	1~240	The coordinate for the bottom boundary of the rectangular motion detection window. The full value range is related to the full image height and 1 is counted from the upper side of the image.

Trigger	0~100	Trigger level of Motion detection.
Sensitivity	0~100	Sensitivity of Motion detection.  ( Sensitivity value:0~100[low~High],set more for more sensitive. )
Color	0000~FFFF	The color of Window area.

## 2.9 Input Port

**Description:** Parameters for hardware input(s).

**CGI:** param.cgi

**Primary group name:** Input

**Primary group value:** I00~

[Input.] [Input.I00.]

Parameter	Value	Description
NbrOfInputs	1~	The number of input.
Name	<String>	The name of the input. (read only)
Trig	open, close	Determines when to trig.
Status	high, low	The status of input.  This parameter is read only.

## 2.10 Output Port

**Description:** Parameters for hardware output(s).

**CGI:** param.cgi

**Primary group name:** Output

**Primary group value:** O00~

[Output .] [Output .O00.]

Parameter	Value	Description
NbrOfOutputs	1~	The number of output.
Name	<String>	The name of the output. (read only)
Active	open, close	Determines when to trig. The active state of the output.
Status	high, low	The status of output.  This parameter is read only.

## 2.11 Network General

**Description:** Network interface settings. The parameters in this group (as opposed to the subgroups of this group) are static network settings. If the Network.BootProto parameter is "dhcp" these parameters may not be in use so always use the read-only parameters in the subgroups to retrieve actual network settings in use by the operating system.

**CGI:** param.cgi

**Primary group name:** Network

**Secondary group name:** ARPPingIPAddress

[Network .] [Network.ARPPingIPAddress.]

Parameter	Value	Description
-----------	-------	-------------

Enabled	yes, no	Enable of ARP Ping IPAddress.
IPAddress	<String>	IP Address.
SubnetMask	<String>	Subnet mask.
DefaultRouter	<String>	Default router/gateway used for connecting devices attached to different networks and network segments.
DNSServer1	<String>	Primary DNS server address.
DNSServer2	<String>	Secondary DNS server address.
BootProto	dhcp, none	<b>none</b> = user assigned IP address. <b>dhcp</b> = get IP from DHCP.

## 2.12 Network Advance

**Description:** Enable/disable retrieval of host name from DHCP-server. The host name, actually in use by the operating system, is located in this group. Enable/disable FTP Server in the product. Enable/disable of Obtain NTP server address via DHCP.

**CGI:** param.cgi

**Primary group name:** Network

[Network .]

Parameter	Value	Description
Resolver.ObtainFromDHCP	yes, no	Enable of Obtain DNS server address via DHCP.
DomainName	<String>	The name entered here will be associated with the product's IP address in the DNS server.
VolatileHostName.ObtainFromDHCP	yes, no	Specifies if the host name should be obtained from a DHCP server.

HostName	<String>	The name of the device on the network, usually the same as the DNS name.
FTP.Enabled	yes, no	Enable the FTP server.

**Primary group name:** Time

[ Time .]

Parameter	Value	Description
ObtainFromDHCP	yes, no	DHCP servers may provide names/IP addresses for local/remote NTP servers. Enable this feature by setting this parameter to yes.
NTP.Server	<String>	Specify the NTP server IP address. Could use domain name.

**Primary group name:** System

[ System .]

Parameter	Value	Description
WebPort	0~65535	HTTP port number.

## 2.13 SMTP(E-Mail)

**Description:** Parameters for the Simple Mail Transfer Protocol, for sending e-mail messages between mail servers, for SMTP authentication.

**CGI:** param.cgi

**Primary group name:** SMTP

**Secondary group name:** Authentication

**Secondary group value:** A01~ A02

[ SMTP .] [ SMTP.Authentication.A01 ]

<b>Parameter</b>	<b>Value</b>	<b>Description</b>
MailServer1	<String>	Primary mail server.
MailServerPort1	0~65535	The port number for the primary mail server.
FromEmail	<String>	E-mail address of the individual or device from which the e-mail is sent.
TestEmail	<String>	E-mail address of the individual or device to test.
Enabled	yes, no	Enable/disable authentication.
UserName	<String>	The user name for the mail server or the POP server.
Password	<String>	The password for the mail server or the POP server.
UsePOP	yes, no	Login to a POP server.
POPServer	<String>	The IP address or host name of the POP server.

## 2.14 DDNS

**Description:** Parameters for the DDNS servers.

**CGI:** param.cgi

**Primary group name:** Network

**Secondary group name:** DDNS

**Secondary group value:** D00~ D01

[ Network.DDNS.D00 ]

Parameter	Value	Description
Enabled	yes, no	Enable of DynDNS service.
ServerAddress	<String>	Specify the DDNS server IP address. Could use domain name.
Port	0~65535	Specify the DDNS server listen port.
RouterPort	0~65535	Specify the IP Sharing device's virtual server listen port.
UpdateTime	600~86400 (Seconds)	Specify the automatic update frequency once start DDNS service.
HostName	<String>	Specify the DynDNS server IP address. Could use domain name.
UserName	<String>	Specify the user name to connect the DynDNS server.
password	<String>	Specify the password to connect the DynDNS server
Response	<String>	This parameter is read only.

## 2.15 PPPoE

**Description:** Parameters for the PPPoE servers.

**CGI:** param.cgi

**Primary group name:** Network

**Secondary group name:** PPPoE

[ Network.PPPoE. ]

Parameter	Value	Description
Enabled	yes, no	Enable / Disable PPPoE server.
UserName	<String>	The user name for the PPPoE server.
Password	<String>	The password for the PPPoE server.

## 2.16 Bonjour

**Description:** Parameters for the Bonjour service.

**CGI:** param.cgi

**Primary group name:** Network

**Secondary group name:** Bonjour

[ Network.Bonjour. ]

Parameter	Value	Description
Enabled	yes, no	Enable / Disable Bonjour service
FriendlyName	<string>	The recognized name shown on the Bonjour service

## 2.17 UPnP

**Description:** Parameters for the UPnP function.

**CGI:** param.cgi

**Primary group name:** Network

**Secondary group name:** UPnP

[ Network.UPnP. ]

Parameter	Value	Description
Enabled	yes, no	Enable / Disable UPnP service
NATTraversal.Enabled	yes, no	Enable / Disable sending port mapping message to UPnP gateway
NATTraversal.HTTPPort	1024 ~ 65535	The gateway port to map to the IP Camera HTTP service port.
NATTraversal.RTSPPort	1024 ~ 65535	The gateway port to map to the IP Camera RTSP over TCP service port.
NATTraversal.RTSPProtocol	“TCP”	It is fixed to “TCP” now.

## 2.18 HTTPS

*Description:* Parameters for HTTPS setting.

*CGI:* param.cgi

*Primary group name:* HTTPS

[HTTPS. ]

Parameter	Value	Description
Enabled	yes, no	https service enabled or not
Port	1024 ~ 65535	Port used by https service. Default value is 443

## 2.19 Wireless

*Description:* Parameters for the Wireless settings.

*CGI:* param.cgi

*Primary group name:* Wireless

[ Wireless. ]

Parameter	Value	Description
Enable	yes, no	Enable of Wireless
Authentication	open, share	WEP key mode.
Channel	1~11	Wireless ad-hoc channel.
Key	0~3	Current WEP key.
Key0	Hex or ASCII	WEP key1
Key1	Hex or ASCII	WEP key2

Key2	Hex or ASCII	WEP key3
Key3	Hex or ASCII	WEP key4
KeyType	“hex”, “ascii”	WEP key type.
Mode	infrastructure, adhoc	Wireless mode.
SSID	<String>	Wireless network ESSID.
Security	none, wep, wpa-psk, wpa2-psk	Wireless security.
Speed	auto, 11g, 11b	Wireless operation mode.
WPEncryption	64, 128 (bit)	WEP Encryption.
WPAEncryption	tkip, aes	WPA Encryption.
WPAKey	ASCII	WPA Key
MACAddress	<String>	Wireless MAC address.  This parameter is <b>read only</b> .
IPAddress	<String>	Wireless IP address.  This parameter is <b>read only</b> .
SubnetMask	<String>	Wireless Subnet Mask.  This parameter is <b>read only</b> .
DefaultRouter	<String>	Wireless Gateway.  This parameter is <b>read only</b> .

## 2.20 *Anonymous login*

**Description:** Enable/disable anonymous login.

**CGI:** param.cgi

**Primary group name:** System

[ System. ]

Parameter	Value	Description
WebProtViewer	anonymous, password	Enable anonymous login. (no user name or password required.)

## 2.21 *Date & Time*

**Description:** Common time information which tell the time zone, how date and time is synchronized and the offset related to the chosen time zone and Coordinated Universal Time, UTC.

**CGI:** param.cgi

**Primary group name:** Time

**Secondary group name:** DST

[Time.][Time.DST. ]

Parameter	Value	Description
Enabled	yes, no	Enable/disable DST (Daylight Saving Time)
SyncSource	PC, NTP, None	The source to synchronize the time with; PC, NTP or None (manually).
ServerDate	A date	The date (yyyy-mm-dd) when the device's time was set manually or synchronized with the computer.
ServerTime	A time	The time (hh:mm:ss) when the device's time was set manually or synchronized with the computer.

TimeZone	GMT-12, ... GMT-1, GMT, GMT+1, ... GMT+12  (Default GMT)	Time zone.
----------	----------------------------------------------------------------------------------	------------

***TimeZone List***

<b>Value</b>	<b>Zone</b>
GMT-12	GMT-12 (Eniwetok, Kwajalein)
GMT-11	GMT-11 (Midway Isl., Samoa)
GMT-10	GMT-10 (Hawaii)
GMT-9	GMT-09 (Alaska)
GMT-8	GMT-08 (Las Vegas, San Francisco, Vancouver)
GMT-7	GMT-07 (Calgary, Denver, Salt Lake City)
GMT-6	GMT-06 (Chicago, Dallas, Mexico City, New Orleans)
Cuba	GMT-05 (Cuba)
GMT-5	GMT-05 (New York, Toronto, Washington DC)
Paraguay	GMT-04 (Paraguay)
Chile	GMT-04 (Chile)
GMT-4	GMT-04 (Charlottetown, Manaus)

Newfoundland	GMT-03:30 (Newfoundland)
GMT-3	GMT-03 (Brazilia, Sao Paulo)
GMT-2	GMT-02 (Noronha, Mid-Atlantic)
GMT-1	GMT-01 (Azores, Cap Verde Isl.)
GMT	GMT (Dublin, Lisbon, London, Reykjavik)
GMT+1	GMT+01 (Amsterdam, Berlin, Rome, Stockholm, Warsaw)
GMT+2	GMT+02 (Athens, Helsinki, Istanbul, Riga)
Egypt	GMT+02 (Egypt)
Israel	GMT+02 (Israel)
Lebanon	GMT+02 (Lebanon)
Syria	GMT+02 (Syria)
GMT+3	GMT+03 (Moscow, Riyadh)
Iraq	GMT+03 (Iraq)
GMT+3,5	GMT+03:30 (Iran)
GMT+4	GMT+04 (Abu Dubai, Baku)
GMT+4,5	GMT+04:30 (Kabul)
GMT+5	GMT+05 (Islamabad, Karachi, Tashkent)
GMT+5,5	GMT+05:30 (Bombay, Calcutta, New Delhi)
GMT+5,75	GMT+05:45 (Kathmandu)

GMT+6	GMT+06 (Novosibirsk, Omsk)
GMT+6,5	GMT+06:30 (Rangoon)
GMT+7	GMT+07 (Bangkok, Hanoi, Jakarta)
GMT+8	GMT+08 (Beijing, Hong Kong, Shanghai, Taipei)
GMT+9	GMT+09 (Osaka, Sapporo, Tokyo, Seoul)
GMT+9,5	GMT+09:30 (Darwin)
Adelaide	GMT+09:30 (Adelaide)
GMT+10	GMT+10 (Hobart)
Sydney	GMT+10 (Canberra, Melbourne, Sydney)
GMT+11	GMT+11 (Solomon Isl.)
GMT+11,5	GMT+11:30 (Norfolk)
GMT+12	GMT+12 (Auckland, Wellington)
Fiji	GMT+12 (Fiji)
Kamchatka	GMT+12 (Kamchatka)
Marshall	GMT+12 (Marshall Isl.)

## 2.22 Language

**Description:** Current language setting.

**CGI:** param.cgi

**Primary group name:** Layout

[ Layout. ]

Parameter	Value	Description
Language	en, tw, cn	Current language.  en = English  tw = 繁體中文 cn = 簡體中文

## 2.23 Log Configuration

**Description:** Parameters for selecting log levels for the product, for log levels to send as e-mail.

**CGI:** param.cgi

**Primary group name:** LogLevels

[ LogLevels. ]

Parameter	Value	Description
CritLogEnabled	yes, no	Enable/disable logging of critical messages.
InfoLogEnabled	yes, no	Enable/disable logging of info messages.
WarningLogEnabled	yes, no	Enable/disable logging of warning messages.

**Primary group name:** MailLogd

[ MailLogd. ]

Parameter	Value	Description
LogSendLevel	0~3	Which messages that are sent with e-mail:  <b>0</b> = None. <b>1</b> = Critical. <b>2</b> = Critical and Warning. <b>3</b> = Critical, Warning and Information.

## 2.24 PTZ

*Description:* Parameters for PTZ control.

*CGI:* param.cgi

*Primary group name:* PTZ

*Primary group value:* C00

*Secondary group name:* Serial

[PTZ.C00. ][PTZ.C00.Serial. ]

Parameter	Value	Description
Enable	0, 1	Enable of PTZ control.  <b>0</b> :Disabled <b>1</b> :Enabled
DeviceID	1~63	PTZ device ID.
Speed	1~10	PTZ speed.
Protocol	PelcoD, PelcoP, Lilin, Dynacolor	PTZ protocol.
PortMode	PTZ, RS232, RS485	Serial port protocol. May be read-only depending on hardware.
BaudRate	300, 1200, 2400, 4800, 9600,	The baudrate used.

	19200, 38400, 57600, 115200	
DataBits	7, 8	The number of data bits.
StopBits	1, 2	The number of stop bits.
Parity	None, Even, Odd	The parity.
Device	/dev/ttyS0, /dev/ttyS1, /dev/ttyS2, /dev/ttyS3 (hardware specific)	The serial device to use.( Default: /dev/ttyS2)  This parameter is read only.

## 2.25 *Layout*

**Description:** Parameters that define how the live view page looks and works.

**CGI:** param.cgi

**Primary group name:** Layout

[Layout . ]

Parameter	Value	Description
Default	yes, no	Enable/disable default layout.
OwnBGColor	black, silver, gray, white, maroon, red, purple, fuchsia, green, lime, olive, yellow, navy, blue, teal, aqua	Selected background color. A string describing a color, readable by the browser.
OwnBGColorEnabled	yes, no	Use own background color.
OwnTextColor	black, silver, gray, white, maroon, red, purple, fuchsia, green, lime, olive, yellow, navy,	Selected text color. A string describing a color, readable by the browser.

	blue, teal, aqua	
OwnTextColorEnabled	yes, no	Use own text color.

**Description:** The properties for custom links at the bottom of the live view page.

**Primary group name:** Layout

**Secondary group name:** CustomLink

**Secondary group value:** C00~C03

[Layout.CustomLink.C00.]

Parameter	Value	Description
Enabled	yes, no	Use custom link #.
Name	<String> (Default Custom link #+1 )	Name on custom link #.
URL	<URL> (Default http:// )	Links to external web pages.

**Description:** Parameters for the background picture on the live view page.

**Primary group name:** Layout

**Secondary group name:** BGPicture

[Layout.BGPicture.]

Parameter	Value	Description
Address	<URL> (Default http:// )	If BGPicture.Source = external, get the image from this address.
Source	none, external (Default none)	Show background picture.  <b>none</b> = no background picture. <b>external</b> = background picture from Layout.BGPictureAddress.

**Primary group name:** Layout

**Secondary group name:** Logo

[Layout.Logo.]

Parameter	Value	Description
Address	<URL> (Default http:// )	If Layout.Logo.Source = external, get image from this address.
Link	none, default, own	Use the logo as a link.  <b>none</b> = no logo link. <b>default</b> = use the default link. <b>own</b> = logo link from Layout.Logo.LinkURL.
LinkURL	<URL> (Default http:// )	URL to where the logo leads if Layout.LogoLink = own.
Source	none, default, own, external	Show logo.  <b>none</b> = no logo. <b>default</b> = default logo. <b>own</b> = logo from Layout.LogoSelect. <b>external</b> = logo from Layout.LogoAddress.

*Primary group name:* Layout

*Secondary group name:* Title

[Layout.Title.]

Parameter	Value	Description
Source	none, default, own	Show title.  <b>none</b> = no title. <b>default</b> = use the default title. <b>own</b> = title from Layout.Title.String.
String	<String> (Default Title )	Title text above image.

*Primary group name:* Layout

**Secondary group name:** Description

[Layout.Description.]

Parameter	Value	Description
Source	none, default, own	Show description text below image.  <b>none</b> = no description. <b>default</b> = use the default description. <b>own</b> = description from Layout.OwnDescription  .String.
String	<String> (Default Description )	Description text below image.

## 2.26 External Video

**Description:** Parameters that define the properties of an external video source.

**CGI:** param.cgi

**Primary group name:** ExternalVideo

**Primary group value:** E00 ~

[ ExternalVideo.E00. ]

Parameter	Value	Description
AudioEnabled	0, 1	Enable of Audio.  <b>0:</b> Disable <b>1:</b> Enable
IPAddress	<String>	IP Address.
Name	<String>	A descriptive external video name.
ProductType	0~7	Product types.

		<b>0:</b> P1400/1401 <b>1:</b> P400/413/423/426/428 <b>2:</b> P461/463(T/DW)/465 <b>3:</b> P8504/4504/4514/4524 <b>4:</b> P500/513/523/525/528 <b>5:</b> P1500/1502 <b>6:</b> P1600 <b>7:</b> P600/606/614/621/623
Resolution	1600x1200, 1600x900, 1280x1024, 1280x960, 1280x720, 1024x768, 1024x576, 800x600, 800x450, 768x576, 768x288, 720x576, 720x480, 704x576, 704x480, 720x288, 720x240, 704x288, 704x240, 640x480, 640x360, 640x240, 480x360, 480x270, 384x288, 352x288, 352x240, 320x240, 320x180, 240x180, 240x135, 192x144, 176x144, 176x120, 176x112, 160x120, 160x90	The image resolution.
Rotation	0~3	Angle in degree clockwise. Whether the image displayed needs rotation or not, as well as the rotation angle.  <b>0</b> = no <b>1</b> = 90 <b>2</b> = 180 <b>3</b> = 270
Stream	0~2	Current stream.
UrlID	<String>	The URL ID for the stream. (Default: <b>stream 1:</b> v00 <b>stream 2:</b> v01 <b>stream 3:</b> v02 )
VideoChannel	0~3	The number of channel.  eg. Channel 1: 0

VideoFormat	H264, mpeg4_rtsp, mjpeg	Video Compression Format.
VideoPort	0~65535	The port number for RTSP.

---

---

## 3 Server Side Include Commands

Server Side Include SSI variables allowed information to be inserted into HTML documents that the server sends to the client.

### 3.1 Pinit

**Description:** It should be placed once before the Pget SSI command.

**Syntax:**

```
<% Pinit(); %>
```

**Example:**

```
<html>
<body>
<% Pinit(); %>
Server IP Address = <% write ( Pget("IPAddress", "Network") ); %>
</body>
</html>
```

**Result (displayed in a Web browser):**

```
Server IP Address = 192.168.0.200
```

### 3.2 Pget

**Description:** It can get a parameter value.

**Syntax:**

```
<% write ( Pget("parameter", "parameter group name") ); %>
```

**Example:**

```
<html>
<body>
<% Pinit(); %>
Stream 1 URL ID = <% write ( Pget("SessionPath", "Image.I00.G00.Stream") ); %>
```

```
</body>  
</html>
```

**Result (displayed in a Web browser):**

```
Stream 1 URL ID = v00
```

### 3.3 Pget\_gpio

**Description:** Get GPIO status.

**Syntax:**

```
<% write ( Pget_gpio ("Status", "Input.I00")); %>
```

**Example:**

```
<html>  
<body>  
<% Pinit(); %>  
Input Port 1 Current State= <% write ( Pget_gpio ("Status", "Input.I00") ); %>  
Input Port 2 Current State= <% write ( Pget_gpio ("Status", "Input.I01") ); %>  
Output Port 1 Current State= <% write ( Pget_gpio ("Status", "Output.O00") ); %>  
</body>  
</html>
```

**Result (displayed in a Web browser):**

```
Input Port 1 Current State= high  
Input Port 2 Current State= high  
Output Port 1 Current State= low
```

### 3.4 Pget\_form

**Description:** It can get the parameter group values.

**Syntax:**

```
<% Pget_form("parameter group name"); %>
```

**Example:**

```
<html>
```

```
<body>
<% Pinit(); %>
<% Pget_form("Time"); %>
</body>
</html>
```

**Result:**

```
<INPUT type=hidden value="yes" name="Time_DST_Enabled">
<INPUT type=hidden value="time.stdtime.gov.tw" name="Time_NTP_Server">
<INPUT type=hidden value="60" name="Time_NTP_Update">
<INPUT type=hidden value="time.stdtime.gov.tw" name="Time_NTP_VolatileServer">
<INPUT type=hidden value="no" name="Time_ObtainFromDHCP">
<INPUT type=hidden value="2009-01-01" name="Time_ServerDate">
<INPUT type=hidden value="14:48:12" name="Time_ServerTime">
<INPUT type=hidden value="None" name="Time_SyncSource">
<INPUT type=hidden value="GMT+8" name="Time_TimeZone">
```

---

---

# *Appendix*

## *A. RTSP URL*

**Description:** Video stream request via RTSP

**Syntax:**

```
rtsp://<server name>/<Stream Session Path>
```

**Example:**

**Stream 1:**

```
rtsp://192.168.0.200/v00
```

**Stream 2:**

```
rtsp://192.168.0.200/v01
```

**Stream 3:**

```
rtsp://192.168.0.200/v02
```

**Note:**

**Default Stream Session Path:**

**Stream 1** = v00

**Stream 2** = v01

**Stream 3** = v02

**Stream Session ID is revisable,**

```
http://<server name>/cgi-bin/admin/param.cgi?actions=update&Image.I0<stream  
number>.G00.Stream.SessionPath=newPath
```

**Get stream session Path:**

```
http://<server name>/cgi-bin/admin/param.cgi?actions=list&Image.I<stream  
number>.G00.Stream.SessionPath
```

***Example:***

**Change Stream1 session Path to v1:**

```
http://192.168.0.200/cgi-bin/admin/param.cgi?actions=update&Image.I00.G00.Stream.SessionPath=v1
```

**Get Stream1 session Path:**

```
http://192.168.0.200/cgi-bin/admin/param.cgi?actions=list&Image.I00.G00.Stream.SessionPath
```

***Return:***

```
Image.I00.G00.Stream.SessionPath=v1
```

## B. Streams Dependency

**Description:** Stream configurations dependency details

### Streams' Availabilities

1. Each stream can be switched to either H.264 or MJPEG mode, except stream 3; it is fixed in H.264 mode only.
2. Each steam can be configured to either CBR or VBR mode.
3. Stream1 (main stream) can chose from all the resolutions; 1600x1200, 1280x720, 800x600, 720x576, 720x480, 640x480, 352x288, 352x240, 320x240.
4. The maximum resolution for Stream2 is 720x576.
5. Stream3 is fixed in the resolution of 352x240.
6. The maximum frame rate for transferring the resolution of 1600x1200 is 15fps. Others can reach up to 30fps.

	STREAM1	STREAM2	STREAM3
<b>Encoding Mode</b>			
<i>H.264 / MJPEG</i>	V	V	V
<b>Transferring Mode</b>			
<i>CBR / VBR</i>	V	V	V
<b>Resolution @ Max. FPS</b>			
<i>1600x1200@15</i>	V		
<i>1280x720@30</i>	V		
<i>800x600@30</i>	V		
<i>720x576@30</i>	V	V	
<i>720x480@30</i>	V	V	
<i>640x480@30</i>	V	V	
<i>352x288@30</i>	V	V	
<i>352x240@30</i>	V	V	
<i>320x240@30</i>	V	V	V

### Streams' Dependency

7. When Stream1 is set to 1600x1200, the frame rate is fixed in 15fps. Both Stream2 and Steam3 can be configured the frame rate from 5 to 15 fps.
8. When Stream1 is set to a resolution smaller than 720x576, say 640x480, Stream 2 can only be set up to 640x480.
9. Frame rate setting for each stream can be independent, for instance, Stream2 can be set 15fps and Stream3 is 30fps while Stream1 (equal or below 1280x720) is set to 5fps.

STREAM1	STREAM2						STREAM3
	720x576	720x480	640x480	352x288	352x240	320x240	320x240 / 352x240 (fixed)
1600x1200, 5~15 fps	5~15 fps						5~15 fps
1280x720, 5~30 fps	5~30 fps						5~30 fps
800x600, 5~30 fps							
720x576, 5~30 fps							
720x480, 5~30 fps							
640x480, 5~30 fps	5~30 fps						
352x240, 5~30 fps	5~30 fps						
352x288, 5~30fps	5~30 fps						
320x240, 5~30 fps	5~30 fps						

Note: The resolutions marked in blue are available in IP Camera (Web UI) only.

## ***C. HTTP status codes***

### ***200 OK***

The request has succeeded. The information returned with the response is dependent on the method used in the request, for example:

GET an entity corresponding to the requested resource is sent in the response;

HEAD the entity-header fields corresponding to the requested resource are sent in the response without any message-body;

POST an entity describing or containing the result of the action;

TRACE an entity containing the request message as received by the end server.

### ***204 No Content***

The server has fulfilled the request but does not need to return an entity-body, and might want to return updated metainformation. The response **MAY** include new or updated metainformation in the form of entity-headers, which if present **SHOULD** be associated with the requested variant.

If the client is a user agent, it **SHOULD NOT** change its document view from that which caused the request to be sent. This response is primarily intended to allow input for actions to take place without causing a change to the user agent's active document view, although any new or updated metainformation **SHOULD** be applied to the document currently in the user agent's active view. The 204 response **MUST NOT** include a message-body, and thus is always terminated by the first empty line after the header fields.

### ***302 Found***

The requested resource resides temporarily under a different URI. Since the redirection might be altered on occasion, the client **SHOULD** continue to use the Request-URI for future requests. This response is only cacheable if indicated by a Cache-Control or Expires header field. The temporary URI **SHOULD** be given by the Location field in the response. Unless the request method was HEAD, the entity of the response **SHOULD** contain a short hypertext note with a hyperlink to the new URI(s).

If the 302 status code is received in response to a request other than GET or HEAD, the user agent **MUST NOT** automatically redirect the request unless it can be confirmed by the user, since this might change the conditions under which the request was issued.

### ***400 Bad Request***

The request could not be understood by the server due to malformed syntax. The client **SHOULD**

**NOT** repeat the request without modifications.

### ***401 Unauthorized***

The request requires user authentication. The response **MUST** include a WWW-Authenticate header field containing a challenge applicable to the requested resource. The client **MAY** repeat the request with a suitable Authorization header field. If the request already included Authorization credentials, then the 401 response indicates that authorization has been refused for those credentials. If the 401 response contains the same challenge as the prior response, and the user agent has already attempted authentication at least once, then the user **SHOULD** be presented the entity that was given in the response, since that entity might include relevant diagnostic information. HTTP access authentication is explained in "HTTP Authentication: Basic and Digest Access Authentication".

### ***404 Not Found***

The server has not found anything matching the Request-URI. No indication is given of whether the condition is temporary or permanent. The 410 (Gone) status code **SHOULD** be used if the server knows, through some internally configurable mechanism, that an old resource is permanently unavailable and has no forwarding address. This status code is commonly used when the server does not wish to reveal exactly why the request has been refused, or when no other response is applicable.

### ***409 Conflict***

The request could not be completed due to a conflict with the current state of the resource. This code is only allowed in situations where it is expected that the user might be able to resolve the conflict and resubmit the request. The response body **SHOULD** include enough information for the user to recognize the source of the conflict. Ideally, the response entity would include enough information for the user or user agent to fix the problem; however, that might not be possible and is not required.

Conflicts are most likely to occur in response to a PUT request. For example, if versioning were being used and the entity being PUT included changes to a resource which conflict with those made by an earlier (third-party) request, the server might use the 409 response to indicate that it can't complete the request. In this case, the response entity would likely contain a list of the differences between the two versions in a format defined by the response Content-Type.

### ***500 Internal Server Error***

The server encountered an unexpected condition which prevented it from fulfilling the request.

### ***501 Not Implemented***

The server does not support the functionality required to fulfill the request. This is the appropriate response when the server does not recognize the request method and is not capable of supporting it for any resource.

### ***502 Bad Gateway***

The server, while acting as a gateway or proxy, received an invalid response from the upstream server it accessed in attempting to fulfill the request.

### ***503 Service Unavailable***

The server is currently unable to handle the request due to a temporary overloading or maintenance of the server. The implication is that this is a temporary condition which will be alleviated after some delay. If known, the length of the delay *MAY* be indicated in a *Retry-After* header. If no *Retry-After* is given, the client *SHOULD* handle the response as it would for a 500 response.