HTTP API

Version 1.23

Update Records:

Number of version	**Date**	**Contents of revision**	**Note**
1.00	April 11, 2011	First Edition	-
1.01	May 16, 2011	4.8 Connection Information for stream **stream0/1/2 -> stream1/2/3**	Important !!!
1.02	May 23, 2011	**4.8 Connection Information for stream** mpeg → mpeg4 **2.3 Motion alarm** Motion cell : (4x3) → (8x6)	
1.03	July 6, 2011	**6. Alarm Out Control** 6.1 Alarm Out(Relay) On/Off Authority is changed. root -> admin Only Administrator can control http/cgi command.	addition Important !!!
1.04	July 18, 2011	4.1 HTTP API → enable function Annex A. Unified Command	addition addition
1.05	September, 20,2011	7.1 JPEG Image Transmission 7.2 MJPEG Image Transmission	addition
1.06	November, 29,2011	7.3 MJPEG Image	addition
1.07	March, 5,2012	4.6 Audio 4.5.1 Send : getmaxsize	Edit addition
1.08	March, 29,2012	3.6 Camera Slow Shutter 5.1 Event "videotamper" → add new parameter	Correct addition
1.09	Jun, 27,2012	3.1 Day&Night Annex A. Unified Command 3	LPR mode addition
1.10	August, 31,2012	2.5 Shock (Video tamper) * shock is usable if supported model. 4.5 Video → capture2.size : Resolution list 4.5 Video → encoder#.source=<string>	addition addition Edit
1.11	November, 21,2012	7.2, 7.3 → "source" parameter 7.3 → "finish" parameter	Addition Edit
1.12	March, 20,2013	1.6 Language 3.7 Camera Shutter Speed 4.7 RTSP – rtsptimelimit value Annex A. Unified Command 3	Addition Edit Edit Edit

1.13	May,2,2013	4.3 Privacy Area	Edit
1.14	Jun,25,2013	4.6 Audio	Correct (Important !!!)
1.15	April,10,2014	1.7 IP address 1.8 DNS	addition
1.16	July,17,20014	1.4 Rebooting (Factory set) 3.2 Color Control (getCapabilitiesImage) Annex B. Get Capabilities (B1.1~B3.2)	Edit addition addition
1.17	November,25,2014	4.2.2 Response tsxton → texton 8. PTZ Control	Fixed addition
1.18	June,15,2015	8.4 Preset/ Autopan/ Tour B3.2 Get Capabilities PTZ	addition addition
1.19	March,22,2016	4.5 Video - need_reboot 7.3	addition deleted
1.20	Novemver,30,2016	2.3 Motion alarm (8x6 → 24x18 blocks) 4.3 Privacy Area (6 / 16) 4.5 Video -Video Resolution	Edit Edit addition
1.21	June,28,2018	5. Event Request eventtype=audiodetect *12M Resolution *H.265 Codec 8.1 PTZ goptzfpos/getptzfpos	addition
1.22	January,28,2020	1.9 Server Model Name 1.10 Server Software Version 3.8 DNR 3.9 Defog 8.5 PTZ Centering	addition
1.23	March,23,2020	4.5 Video Annex B	Edit

[Contents]

1. System Configuration

2. Event Configuration

3. Camera Configuration

4. Stream Configuration

5. Event Request

6. Alarm Out Control

7. Image Transmission

8. PTZ Control

Annex A. Unified Command

Annex B. Get Capabilities

* What is included here may vary depending on the model.
Also, the features provided may not match the administrator settings included with the product.
We recommend that you check and use by product.

1. System Configuration

1.1 Server Name

```
  http://<servername>/cgi-bin/control/servername.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getservername setservername	get : Get curret value set : Set value using given data.
servername= <string>	10-charactors	English only

1.1.1 Send --Method: GET

Example 1. Get / Set server Name to “welcome”

```
http://192.168.1.30/cgi-bin/control/servername.cgi?id=admin&passwd=admin
&action=getservername
http://192.168.1.30/cgi-bin/control/servername.cgi?id=admin&passwd=admin
&action=setservername&servername=welcome
```

1.1.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
servername = welcome
```

<**If Error**>

```
Error: <description>
```

1.2 Server Date

```
  http://<servername>/cgi-bin/control/serverdate.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getdate setdate	get : Get curret value set : Set value using given data.
year=<int>	1970~2031	
month=<int>	1~12	
day=<int>	1~31	
hour=<int>	0~23	
minute=<int>	0~59	
second=<int>	0~59	Optional. if it is not set, set to 0.

* It will be change date/time mode to user.

1.2.1 Send   --Method: GET

Example 1. Set server date to "2011-03-08 12:30"

```
http://192.168.1.30/cgi-bin/control/serverdate.cgi?id=admin&passwd=admin
&action=getdate
http://192.168.1.30/cgi-bin/control/serverdate.cgi?id=admin&passwd=admin
&action=setdate&year=2011&month=3&day=8&hour=12&minute=30
```

1.2.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
serverdate = <month> <day>, <year> <hour>:<minute>:<second>
```

<**If Error**>

```
Error: <description>
```

1.3 MAC

```
  http://<servername>/cgi-bin/control/servermac.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getmac	get : Get curret value

1.3.1 Send --Method: GET

Example 1. Get MAC / Set MAC

```
http://192.168.1.30/cgi-bin/control/servermac.cgi?id=admin&passwd=admin
&action=getmac
```

1.3.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
mac = xx:xx:xx:xx:xx:xx
```

<**If Error**>

```
Error:<description>
```

1.4 Rebooting

```
  http://<servername>/cgi-bin/control/reboot.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	setreboot setfactory setfactoryexip	setreboot : Just reboot only. setfactory : set default value setfactoryexip : set default, except IP/DNS/HTTP Port

1.4.1 Send --Method: GET

Example 1. Get MAC / Set MAC

```
http://192.168.1.30/cgi-bin/control/reboot.cgi?id=admin&passwd=admin
&action=setreboot
http://192.168.1.30/cgi-bin/control/reboot.cgi?id=admin&passwd=admin
&action=setfactory
http://192.168.1.30/cgi-bin/control/reboot.cgi?id=admin&passwd=admin
&action=setfactoryexip
```

1.4.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
reboot = rebooting
factoryset = rebooting
factorysetexip = rebooting
```

<**If Error**>

```
Error:<description>
```

1.5 Web port

```
http://<servername>/cgi-bin/control/webport.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	setwebport	set : Set value using given data.
webport=<int>	80,	3000~60000

1.5.1 Send --Method: GET

Example 1. Set Web port

```
http://192.168.1.30/cgi-bin/control/webport.cgi?id=admin&passwd=admin
&action=setwebport&webport=80
```

1.5.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
webport = 80
```

<**If Error**>

```
Error:<description>
```

1.6 Language

```
http://<servername>/cgi-bin/control/language.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getlang setlang	get : Get curret value set : Set value using given data.
language=<string>	english	korean, english, polish, russian,persian

1.6.1 Send --Method: GET

Example 1. Set Language

```
http://192.168.1.30/cgi-bin/control/language.cgi?id=admin&passwd=admin
&action=setlang&language=english
```

1.6.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
language = english
```

<**If Error**>

```
Error:<description>
```

1.7 IP address

```
 http://<servername>/cgi-bin/control/netset.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getip setip	get : Get curret value set : Set value using given data.
mode=<string>	static dhcp	
ipaddress=<string>		IPv4 (xxx.xxx.xxx.xxx)
netmask=<string>		IPv4 (xxx.xxx.xxx.xxx)
gateway=<string>		IPv4 (xxx.xxx.xxx.xxx)

1.7.1 Send --Method: GET

Example 1. Set Language

```
http://192.168.1.30/cgi-bin/control/netset.cgi?id=admin&passwd=admin
&action=setip&mode=static&ipaddress=192.168.1.31&netmask=255.255.255.0
&gateway=192.168.1.1
```

1.7.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
mode = static
ipaddress = 192.168.1.31
netmask = 255.255.255.0
gateway = 192.168.1.1
```

```
mode = dhcp
```

<**If Error**>

```
Error:<description>
```

1.8 DNS

```
  http://<servername>/cgi-bin/control/dnsset.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getdns setdns	get : Get curret value set : Set value using given data.
firstdns=<string>		IPv4 (xxx.xxx.xxx.xxx)
seconddns=<string>		IPv4 (xxx.xxx.xxx.xxx)

1.8.1 Send --Method: GET

Example 1. Set Language

```
http://192.168.1.30/cgi-bin/control/dnsset.cgi?id=admin&passwd=admin
&action=setdns& firstdns=168.126.63.1
```

1.8.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
firstdns = 168.126.63.1
seconddns =
```

<**If Error**>

```
Error:<description>
```

1.9 Server Model Name

```
  http://<servername>/cgi-bin/control/servermodel.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getservermodel	get : Get curret value

1.9.1 Send --Method: GET

Example 1. Get Server Model name

```
http://192.168.1.30/cgi-bin/control/servermodel.cgi?id=admin&passwd=admin
&action=getservermodel
```

1.9.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
ServerModel = XXXX
```

<**If Error**>

```
Error: <description>
```

1.10 Server Software Version

```
  http://<servername>/cgi-bin/control/versioninfo.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getversioninfo	get : Get curret value

1.10.1 Send --Method: GET

Example 1. Get server Software version information

```
http://192.168.1.30/cgi-bin/control/versioninfo.cgi?id=admin&passwd=admin
&action=getversioninfo
```

1.10.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Version]
Hardware = XXXX
Firmware = XXXX
ActiveX = XXXX
HTTP/CGI = XXXX
ONVIF = XXXX
```

<**If Error**>

```
Error: <description>
```

2. Event Configuration

2.1 Alarm Input

```
http://<servername>/cgi-bin/control/alarmin.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getalarmin# setalarmin#	get : Get curret value set : Set value using given data. # is Alarm-in Number. Current, only support 1
allstatus= <string>	enable, disable	
alarmin#.enable= <string>	enable, disable	
alarmin#.name= <string>	10-charactors	
alarmin#.type= <string>	nc no	normal open / normal close

2.1.1 Send --Method: GET

Example 1. Get/Set alarm input 1 enable, name sensor, type no

```
http://192.168.1.30/cgi-bin/control/alarmin.cgi?id=admin&passwd=admin
&action=getalarmin1
http://192.168.1.30/cgi-bin/control/alarmin.cgi?id=admin&passwd=admin
&action=setalarmin1&allstatus=enable&alarmin1.enable=enable&alarmin1.name=sensor
&alarmin1.type=no
```

2.1.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
allstatus = enable
maxno = 1            * Max supported number of alarm input

alarmin1.enable = enable
alarmin1.name = sensor
alarmin1.type = no
```

<**If Error**>

```
Error:<description>
```

2.2 Alarm Output

http://<*servername*>/cgi-bin/control/alarmout.cgi
[?<*argument*>=<*value*>[&<*argument*>=<*value*>...]]

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getalarmout# setalarmout#	get : Get curret value set : Set value using given data. # is Alarm-out Number. Current, only support 1 if use "getalarmout0", send all of alarm output.
allstatus= <string>	enable, disable	
alarmout#.enable= <string>	enable, disable	
alarmout#.name= <string>	10-charactors	
alarmout#.link= <int>	0~3 or 0~7	motion(1)|alarminput(2)|shock(4) * "|" is bit-OR. * shock is usable if supported model.
alarmout#.time= <int>	1/5~180	1:continuous, 5~180: seconds

2.2.1 Send --Method: GET

Example 1. Get/Set alarm output 1 enable, link alarmin+motion, duration time 5

```
http://192.168.1.30/cgi-bin/control/alarmout.cgi?id=admin&passwd=admin
&action=getalarmout1
http://192.168.1.30/cgi-bin/control/alarmout.cgi?id=admin&passwd=admin
&action=setalarmout1&allstatus=enable&alarmout1.enable=enable&alarmout1.name=relay
&alarmout1.link=3&alarmout1.time=5
```

2.2.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
allstatus = enable
maxno = 1               * Max supported number of alarm output

alarmout1.enable = enable
alarmout1.name = relay
alarmout1.link = 3
alarmout1.time = 5   *continuous, 5~180
```

<**If Error**>

```
Error:<description>
```

2.3 Motion event

```
  http://<servername>/cgi-bin/control/motion.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getmotion# setmotion#	get : Get curret value set : Set value using given data. # is Motion Number. Current, only support 1
allstatus= <string>	enable, disable	
mdduration= <int>	0~10	motion event duration time(second). if supported model only.
mdtimeoff= <int>	1/2	1:Continuous 2:Timeoff (Event Off after duration time)
motion#.enable= <string>	enable, disable	
motion#.name= <string>	10-charactors	
motion#.level= <int>	1~5	
motion#.size= <string>	24x18	24x18 :18-row 24-column , total 432 block
motion#.area1= <int>	1|2|4|8|16|32|64| 128 ...	* “|” is bit-OR 0~16777215 (0xffffff)
motion#.area2= <int>	1|2|4|8|16|32|64| 128 ...	
motion#.area3= <int>	1|2|4|8|16|32|64| 128 ...	
motion#.area4= <int>	1|2|4|8|16|32|64| 128 ...	
motion#.area5= <int>	1|2|4|8|16|32|64| 128 ...	
motion#.area6= <int>	1|2|4|8|16|32|64| 128 ...	
:		
motion#.area18= <int>	0x000000~0xffff ff	

- Area division : 24x18 size

area1	1	2	4	8	16	32	...	0x400000	0x800000
area2	1	2	4	8	16	32	...	0x400000	0x800000
area3	1	2	4	8	16	32	...	0x400000	0x800000
area4	1	2	4	8	16	32	...	0x400000	0x800000
area5	1	2	4	8	16	32	...	0x400000	0x800000
:	:	:	:	:	:	:	:	:	:
area17	1	2	4	8	16	32	...	0x400000	0x800000
area18	1	2	4	8	16	32	...	0x400000	0x800000

2.3.1 Send --Method: GET

Example 1. Get/Set motion1 enable, name md1, level 5

```
http://192.168.1.30/cgi-bin/control/motion.cgi?id=admin&passwd=admin
&action=getmotion1
http://192.168.1.30/cgi-bin/control/motion.cgi?id=admin&passwd=admin
&action=setmotion1&motion1.level=3

http://192.168.1.30/cgi-bin/control/motion.cgi?id=admin&passwd=admin
&action=getmotion1&motion1.size=24x18

http://192.168.1.30/cgi-bin/control/motion.cgi?id=admin&passwd=admin
&action=setmotion1&allstatus=enable&mdduration=3&motion1.enable=enable&motio
n1.name=md1&motion1.level=3&motion1.size=24x18
&motion1.area1=1&motion1.area2=2&motion1.area3=4&motion1.area4=8&motion1.are
a5=16&motion1.area6=32&motion1.area7=64&motion1.area8=128&motion1.area9=256
&motion1.area10=512&motion1.area11=1024&motion1.area12=2048&motion1.area13=
4096&motion1.area14=8192&motion1.area15=16384&motion1.area16=32768&motion1.
area17=65536&motion1.area18=131072
```

2.3.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
allstatus = enable
maxno = 1             * Max supported number of motion.

motion1.enable = enable
motion1.name = md1
motion1.level = 3
motion1.size = 24x18
motion1.area1 = 0
motion1.area2 = 15
motion1.area3 = 15
motion1.area4 = 15
motion1.area5 = 15
motion1.area6 = 0
```

```
:
motion1.area18 = 131072        * mdsize is 24x18
```

<**If Error**>

```
Error:<description>
```

2.4 Record

```
  http://<servername>/cgi-bin/control/recordevent.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getrecevent setrecevent	get : Get curret value set : Set value using given data. # is Motion Number. Current, only support 1
record.status= <string>	enable, disable	
record.streamid= <string>	stream1 stream2 stream3	
record.link= <int>	0~3 or 0~7	motion(1) | alarm input(2) | shock(4) * "|" is bit-OR * shock is usable if supported model.
record.save <int>	0/1/2/3	ftp-upload(1) | email-attach(2) * "|" is bit-OR
record.timeprev= <int>	0~5	record time before event
record.timenext= <int>	5~30	record time after event
record.maxsize= <int>	4096~10240	Max file size for recording(4M~10M)

2.4.1 Send --Method: GET

Example 1. Get/Set record enable, streamed:stream1 link:motion|alarm-input save:ftp
previous time:5 next time:20 maxsize:2Mbyte

```
http://192.168.1.30/cgi-bin/control/recordevent.cgi?id=admin&passwd=admin
&action=getrecevent
http://192.168.1.30/cgi-bin/control/recordevent.cgi?id=admin&passwd=admin
&action=setrecevent&record.status=enable&record.streamid=stream1&record.link=3
&record.save=1&record.timeprev=5&record.timenext=20&record.maxsize=2048
```

2.4.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
record.status = enable
record.streamid = stream1
record.link = 3
record.save = 1
record.timeprev = 5
record.timenext = 20
record.maxsize = 2048
```

<**If Error**>

```
Error:<description>
```

3. Camera Configuration

3.1 Day & Night

```
http://<servername>/cgi-bin/control/camdaynight.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getdaynight setdaynight	get : Get curret value set : Set value using given data.
mode=<string>	day night auto lpr	LPR Mode * if supported model only
interval=<int>	1~200	D&N change Interval(Second)
ptn=<int>	1~990	Point to night ← day Point to night → day ptn >= ptd
ptd=<int>		
irlink=<string>	enable disable	

3.1.1 Send --Method: GET

Example 1. Get/Set day&night mode auto, autolevel 50, irlink enable

```
http://192.168.1.30/cgi-bin/control/camdaynight.cgi?id=admin&passwd=admin
&action=getdaynight
http://192.168.1.30/cgi-bin/control/camdaynight.cgi?id=admin&passwd=admin
&action=setdaynight&mode=auto&interval=5&irlink=enable&ptn=500&ptd=270
```

3.1.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
daynightmode = auto
interval = 5
ptn = 15
ptd = 38
irlink = enable
```

<**If Error**>

```
Error:<description>
```

3.2 Color Control

```
  http://<servername>/cgi-bin/control/camcolor.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getcolor setcolor getncolor setncolor getCapabilitiesImage	get : Get curret value set : Set value using given data. getn: Get value for night. setn: Set value for night. *if supported get: Get Capability
bright=<int>	1-100	
contrast=<int>	1-100	
saturation=<int>	1-100	
sharp=<int>	1-100	
edge=<int>	1-100	* edge enhancement
hue=<int>	1-100	*

* : supported model only

3.2.1 Send   --Method: GET

Example 1. Get/Set color bright 52, constrst 5, saturation 100, sharp 100, edge 68 or hue 50

```
http://192.168.1.30/cgi-bin/control/camcolor.cgi?id=admin&passwd=admin
&action=getcolor

http://192.168.1.30/cgi-bin/control/camcolor.cgi?id=admin&passwd=admin
&action=setcolor&bright=52&contrast=5&saturation=100&sharp=100&edge=68

http://192.168.1.30/cgi-bin/control/camcolor.cgi?id=admin&passwd=admin
&action=setcolor&hue=50

http://192.168.1.30/cgi-bin/control/camcolor.cgi?id=admin&passwd=admin
&action=getCapabilitiesImage
```

3.2.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
bright = 52
contrast = 5
saturation = 100
sharp = 100
edge = 68   *
hue = 50    *
```

<**If Success : getCapabilitiesImage** >

```
[ImageEnhanceSupported]
brightness = Yes
contrast = Yes
saturation = Yes
sharpness = Yes
edgeenhance = Yes
hue = No

[ImageEnhanceValuerange]
brightness = min:1;max:100;default:50
contrast = min:1;max:100;default:50
saturation = min:1;max:100;default:50
sharpness = min:1;max:100;default:50
edgeenhance = min:1;max:100;default:50
hue = min:1;max:100;default:50
```

<**If Error**>

```
Error:<description>
```

3.3 White balance

```
http://<servername>/cgi-bin/control/camwhitebal.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getwb setwb	get : Get curret value set : Set value using given data.
mode=<string>	auto indoor outdoor fluorescent user push autoindoor	Each item is supported according to the model.
userred=<int>	1-100	Optional if set "mode=user", you must use this.
userblue=<int>	1-100	Optional if set "mode=user", you must use this.

3.3.1 Send --Method: GET

Example 1. Get/Set white balance mode user, user red 50 user blue 50

```
http://192.168.1.30/cgi-bin/control/camwhitebal.cgi?id=admin&passwd=admin
&action=getwb
http://192.168.1.30/cgi-bin/control/camwhitebal.cgi?id=admin&passwd=admin
&action=setwb&mode=user&userred=50&userblue=50
```

3.3.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
wbmode = user
userred = 50
userblue = 50
```

<**If Error**>

```
Error:<description>
```

3.4 WDR

```
 http://<servername>/cgi-bin/control/camwdr.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getwdr setwdr	get : Get curret value set : Set value using given data.
wdrstatus=<string>	enable disable	
mode=<string>	compensation dwdr	
compensationmode =<string>	front back	Optional if set "mode=compensation", you must use this.
dwdrmode=<string>	step1~step5	Optional if set "mode=dwdr", you must use this.

3.4.1 Send --Method: GET

Example 1. Get/Set wdr enable, mode dwdr, dwdrmode step1

```
http://192.168.1.30/cgi-bin/control/camwdr.cgi?id=admin&passwd=admin
&action=getwdr
http://192.168.1.30/cgi-bin/control/camwdr.cgi?id=admin&passwd=admin
&action=setwdr&wdrstatus=enable&mode=dwdr&dwdrmode=step1
```

3.4.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
wdrstatus = enable
mode = dwdr
dwdrmode = step1
```

<**If Error**>

```
Error:<description>
```

3.5 Camera Effect

```
http://<servername>/cgi-bin/control/cameffect.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	geteffect seteffect	get : Get curret value set : Set value using given data.
colorbar=<string>	enable disable	
monoimg=<string>	enable disable	mono-chrome image
negative=<string>	enable disable	negative image

3.5.1 Send --Method: GET

Example 1. Get/Set effect colorbar disable , monoimg disable, negative disable

```
http://192.168.1.30/cgi-bin/control/cameffect.cgi?id=admin&passwd=admin
&action=geteffect
http://192.168.1.30/cgi-bin/control/cameffect.cgi?id=admin&passwd=admin
&action=seteffect&colorbar=disable&monoimg=disable&negative=disable
```

3.5.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
colorbar = disable
monoimg = disable
negative = disable
```

<**If Error**>

```
Error:<description>
```

3.6 Camera Slow Shutter

```
http://<servername>/cgi-bin/control/camslowshut.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getslowsh setslowsh	get : Get curret value set : Set value using given data.
slowshutstatus=<string>	enable disable	
slowshutter=<int>	1-100	

3.6.1 Send --Method: GET

Example 1. Get/Set slow shutter value 14

```
http://192.168.1.30/cgi-bin/control/camslowshut.cgi?id=admin&passwd=admin
&action=getslowsh
http://192.168.1.30/cgi-bin/control/camslowshut.cgi?id=admin&passwd=admin
&action=setslowsh&slowshutstatus=enable&slowshutter=14
```

3.6.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
slowshutstatus = enable
slowshutter = 2
```

<**If Error**>

```
Error:<description>
```

3.7 Camera Shutter Speed

```
http://<servername>/cgi-bin/control/camshutspeed.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getshutterspd setshutterspd	get : Get curret value set : Set value using given data.
shutmode=<string>	auto suppressroll user	shutter mode suppressroll : 50/60Hz lock. user: shutter speed set by user.
maxexposure=<int>	1/2/3/4/5/6	NTSC: 30/60/120/180/240/auto PAL : 25/50/100/150/200/auto if "shutmode" is "auto" * Supported Model only.
suppress=<string>	week strong	Optional if set "shutmode=suppressroll", you must set this.
shutspeed=<int>	1/2/3/4/5/6/7/8/9	NTSC: 15/30/60/120/240/480/960/4800/9000 PAL : 12.5/25/50/100/200/400/800/4000/8000 1/12.5~1/9000 second Optional if set "shutmode=user", you must set this.
agcvalue=<int>	1-100	Optional if set "shutmode=user", you must set this.

3.7.1 Send --Method: GET

Example 1. Get/Set shutter speed mode user, shutter speed 1/9000, agc 60

```
http://192.168.1.30/cgi-bin/control/camshutspeed.cgi?id=admin&passwd=admin
&action=getshutterspd
http://192.168.1.30/cgi-bin/control/camshutspeed.cgi?id=admin&passwd=admin
&action=setshutterspd&shutmode=user&shutspeed=9&agcvalue=60
```

3.7.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
shutmode = user
shutspeed = 9
```

```
agcvalue = 60
minshutspd = 6              * if shutmode is set auto, it will be display.
```

<**If Error**>

```
Error:<description>
```

3.8 DNR

```
 http://<servername>/cgi-bin/control/camdnr.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getdnr setdnr	get : Get curret value set : Set value using given data.
dnstatus=<string>	enable disable	
mode=<string>	dnr2d dnr3d	
dnrvalue=<int>	1~100	
dynamic=<string>	enable disable	Supported model only

3.8.1 Send --Method: GET

Example 1. Get/Set wdr enable, mode dwdr, dwdrmode step1

```
http://192.168.1.30/cgi-bin/control/camdnr.cgi?id=admin&passwd=admin
&action=getdnr
http://192.168.1.30/cgi-bin/control/camdnr.cgi?id=admin&passwd=admin
&action=setdnr&dnrstatus=enable&mode=dnr3d&dnrvalue=50
```

3.8.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
dnrstatus = enable
mode = dnr3d
dnrvalue = 50
dynamic = enable
```

<**If Error**>

```
Error:<description>
```

3.9 Defog

http://<*servername*>/cgi-bin/control/camdefog.cgi
[?<*argument*>=<*value*>[&<*argument*>=<*value*>...]]

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getdefog setdefog	get : Get curret value set : Set value using given data
defogen=<string>	enable disable	
mode=<string>	auto manual	auto : Supported model only.
defogvalue=<int>	1~100	Supported model only.

3.9.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/camdefog.cgi?id=admin&passwd=admin
&action=getdefog

http://192.168.1.30/cgi-bin/control/camdefog.cgi?id=admin&passwd=admin
&action=setdefog&defogen=enable&mode=auto&defogvalue=80
```

3.9.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
defogstatus = enable
mode = auto
defogvalue = 80
```

<**If Error**>

```
Error: <description>
```

4. Stream Configuration

4.1 HTTP API

```
 http://<servername>/cgi-bin/control/httpapi.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	setapi	set : Set value using given data.
apictrlstatus=<string>	enable disable	http Api for controls

4.1.1 Send --Method: GET

Example 1. Set http api disable

```
http://192.168.1.30/cgi-bin/control/httpapi.cgi?id=admin&passwd=admin
&action=setapi&apictrlstatus=disable
```

4.1.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
apictrlstatus = disable
```

<**If Error**>

```
Error:<description>
```

4.2 OSD

```
http://<servername>/cgi-bin/control/osd.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getosd setosd	get : Get curret value set : Set value using given data.
osdstatus=<string>	enable disable	osd all display
texton=<string>	enable disable	test string
textstr=<string>	char 10	now, only support english
dateon=<string>	enable disable	date display
datetype=<string>	yyyy-mm-dd dd-mm-yyyy mm-dd-yyyy	date display type year, month, day
timeon=<string>	enable disable	time display
eventon=<string>	enable disable	event display
eventdnon=<string>	enable disable	day&night
eventmdon=<string>	enable disable	motion detection
eventsenson=<string>	enable disable	alarm input(sensor)
eventrelayon=<string>	enable disable	alarm out(relay)
eventshock=<string>	enable disable	shock(video tamper) * shock is usable if supported model.
osdfont=<string>	english	language font
osdsize=<string>	big small	font size
osdcolor=<string>	white	font color. now, only support white.

4.2.1 Send --Method: GET

Example 1. Get/Set osdstatus enable, texton enable, txtstr camera, font English, size big

```
http://192.168.1.30/cgi-bin/control/osd.cgi?id=admin&passwd=admin
&action=getosd
http://192.168.1.30/cgi-bin/control/osd.cgi?id=admin&passwd=admin
&action=setosd&osdstatus=enable&texton=enable&textstr=camera&osdfont=English
&osdsize=big
```

4.2.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
osdstatus = enable
texton = enable
textstr = camera
dateon = enable
datetype = yyyy-mm-dd
timeon = enable
eventon = enable
eventdnon = enable
eventmdon = enable
eventsenson = enable
eventrelayon = enable
eventshock = enable          *shock will be displayed if supported model.
osdfont = english
osdsize = big
osdcolor = white
```

<**If Error**>

```
Error:<description>
```

4.3 Privacy Area

http://<*servername*>/cgi-bin/control/privacy.cgi
[?<*argument*>=<*value*>[&<*argument*>=<*value*>...]]

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getprivacy# setprivacy#	get : Get curret value set : Set value using given data. # is Alarm-out Number. (1~6 / 1~16)
allstatus=<string>	enable disable	
privacy.enable=<string>	enable disable	
privacy.color=<int>	1~8	white ~ black
privacy.startx=<int>	0~319	start x
privacy.starty=<int>	0~239	start y
privacy.endx=<int>	1~320	width. startx+endx < 320
privacy.endy=<int>	1~240	height. starty+endy < 240

4.3.1 Send --Method: GET

Example 1. Get/Set privacy1 enable, color white, startx 0, starty 0, endx 30, endy 30

```
http://192.168.1.30/cgi-bin/control/privacy.cgi?id=admin&passwd=admin
&action=getprivacy1
http://192.168.1.30/cgi-bin/control/privacy.cgi?id=admin&passwd=admin
&action=setprivacy1&allstatus=enable&privacy.enable=enable&privacy.color=1
&privacy.startx=0&privacy.starty=0&privacy.endx=30&privacy.endy=30
```

4.3.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
allstatus = enable
maxno = 6

privacy1.enable = enable
privacy1.color = 1
privacy1.startx = 0
privacy1.starty = 0
privacy1.endx = 30
privacy1.endy = 30
```

<**If Error**>

```
Error:<description>
```

4.4 TV OUT

```
  http://<servername>/cgi-bin/control/tvout.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	gettvout settvout	get : Get curret value set : Set value using given data.
tvoutstatus=<string>	enable disable	
tvtype=<string>	ntsc pal	

4.4.1 Send --Method: GET

Example 1. Get/Set tvout status enable, type ntsc

```
http://192.168.1.30/cgi-bin/control/tvout.cgi?id=admin&passwd=admin
&action=gettvout
http://192.168.1.30/cgi-bin/control/tvout.cgi?id=admin&passwd=admin
&action=settvout&tvoutstatus=enable&tvtype=ntsc
```

4.4.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
tvoutstatus = enable
tvtype = ntsc
```

<**If Error**>

```
Error:<description>
```

4.5 Video

```
http://<servername>/cgi-bin/control/videoset.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getvideo setvideo# getmaxsize	get : Get curret value set : Set value using given data. where # is encoder number.(1~3) getmaxsize: Get Max video size
videoflip=<string>	normal mirror flip both	both : flip+mirror
capture1.enable=<string>	enable disable	
capture1.size=<string>	4000x3000 3000x3000 4096x2160 3840x2160 2560x1920 2560x1600 2560x1440 2048x1536 1600x1200 1920x1080 1280x1024 1280x960 1280x720 1024x768	video size 12M: 4000x3000 → 9M: 3000x3000 → 8M(4K) : 4096x2160 → 5M: 2560x1920 → 3M: 2048x1536 → 2M: 1920x1080 → 1M: 1280x1024 →
capture2.enable=<string>	enable disable	
capture2.size=<string>	2048x1536 1600x1200 1920x1080 1280x1024 1280x960 1280x720 1024x768 854x480 640x480 320x240 nouse	video size *This may vary depending on a model.
capture3.enable=<string>	enable disable	* if supported
capture3.size=<string>	2560x1920 2560x1600 2048x1536 1600x1200 1920x1080 1280x1024 1280x960 1280x720 1024x768 640x480	video size *This may vary depending on a model.

	320x240 nouse	
encoder#.enable	enable disable	# is encoder number.
encoder#.source=<string>	capture1 capture2 capture3 :	Source encoder1 : only capture1
encoder#.codec=<string>	h265 h264 mpeg mjpeg	h265: H.265 h264: H.264 mpeg: MPEG4 mjpeg: MJPEG
encoder#.framerate=<int>	30	frame rate. upto 30/60 60 : Supported model only
encoder#.quality=<int>	4000 50	H265,h264,mpeg: bitrate (kbps) (128~8192/12000) mjpeg : Q value(5~95)
encoder#.keyframe=<int>	30	it is used only h264 or mpeg (10~60)
encoder#.ratectrl=<string>	cbr vbr	bit-rate control

4.5.1 Send --Method: GET

Example 1. Get/Set

```
http://192.168.1.30/cgi-bin/control/videoset.cgi?id=admin&passwd=admin
&action=getvideo
```

```
http://192.168.1.30/cgi-bin/control/videoset.cgi?id=admin&passwd=admin
&action=setvideo1&encoder1.enable=enable&encoder1.source=capture1
&encoder1.codec=h264&encoder1.framerate=15&encoder1.quality=512
&encoder1.keyframe=30&encoder1.ratectrl=vbr
```

```
http://192.168.1.30/cgi-bin/control/videoset.cgi?id=admin&passwd=admin
&action=getmaxsize
```

4.5.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
videoflip = normal
```

```
capture1.enable = enable
capture1.size = 1280x960
capture2.enable = enable
capture2.size = 640x480
capture3.enable = enable
capture3.size = 320x240

encoder1.enable = enable
encoder1.source = capture1
encoder1.codec = h264
encoder1.framerate = 30
encoder1.quality = 512
encoder1.keyframe = 30
encoder1.ratectrl = vbr

encoder2.enable = enable
encoder2.source = capture2
encoder2.codec = mpeg
encoder2.framerate = 30
encoder2.quality = 512
encoder2.keyframe = 30
encoder2.ratectrl = vbr

encoder3.enable = enable
encoder3.source = capture3
encoder3.codec = mjpeg
encoder3.framerate = 30
encoder3.quality = 50

need_reboot = yes              * yes/no. if yes, it is applied after reboot.
```

```
max_video_size = 1280x1024
```

<**If Error**>

```
Error:<description>
```

4.6 Audio

```
http://<servername>/cgi-bin/control/audioset.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getaudio setaudio	get : Get curret value set : Set value using given data.
audiocodec=<string>	ulaw alaw	now, only support ulaw
audioinenable=<string>	enable disable	audio input
audioingain=<int>	1~100	audio input volume
audiooutenable=<string>	enable disable	audio output
audiooutgain=<int>	1~100	audio output volume
audiosampling=<int>	8000/16000	audion sampling rate

4.6.1 Send --Method: GET

Example 1. Get/Set codec ulaw, audio in enable, audio out disable

```
http://192.168.1.30/cgi-bin/control/audioset.cgi?id=admin&passwd=admin
&action=getaudio
http://192.168.1.30/cgi-bin/control/audioset.cgi?id=admin&passwd=admin
&action=setaudio&audiocodec=ulaw&audioinenable=enable&audiooutenable=disable
```

4.6.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
audiocodec = ulaw
audioout = disable
audiooutgain = 50
audioin = enable
audioingain = 50
audiosampling = 8000
```

<**If Error**>

```
Error:<description>
```

4.7 RTSP

```
http://<servername>/cgi-bin/control/rtspset.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getrtsp setrtsp	get : Get curret value set : Set value using given data.
rtspport=<int>	554	554, 3000~60000
rtpport=<int1>,<int2>	5000,5999	3000~60000, range: <int1> ~ <int2>
rtcpenable=<string>	enable disable	
rtsptimelimit=<int>	~~0~30~~ 0, 60~300	rtsp connection is released after set time. 0: OFF 60~300:Seconds
multicastenable=<string>	enable disable	
multicastttl=<int>	5	1~128
multicastvideoip=<string>	239.249.119.123	D class IP address
multicastvideoport=<int>	40000	3000~60000
multicastaudioip=<string>	239.249.119.129	D class IP address
multicastaudioport=<int>	40002	3000~60000
authorityenable=<string>	enable disable	user confirm

4.7.1 Send --Method: GET

Example 1. Get/Set rtspport 554, rtp port 5000~5999

```
http://192.168.1.30/cgi-bin/control/rtspset.cgi?id=admin&passwd=admin
&action=getrtsp
http://192.168.1.30/cgi-bin/control/rtspset.cgi?id=admin&passwd=admin
&action=setrtsp&rtspport=554&rtpport=5000,5999
```

4.7.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
rtspport = 554
rtpport = 5000,5999
rtcpenable = enable
rtsptimelimit = enable
multicastenable = disable
multicastttl = 5
multicastvidioip = 239.249.119.123
multicastvidioport = 40000
multicastaudioip = 239.249.119.129
multicastaudioport = 4002
authorityenable = enable
```

<**If Error**>

```
Error:<description>
```

4.8 Connection Information for stream

```
  http://<servername>/cgi-bin/control/connectinfo.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getconnect	get : Get curret value
stream=<string>	all	

4.8.1 Send --Method: GET

Example 1. Get information all

```
http://192.168.1.30/cgi-bin/control/connectinfo.cgi?id=admin&passwd=admin
&action=getconnect&stream=all
```

4.8.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
stream1.codec = h.265
stream1.size = 1920x1080
stream1.address = rtsp://ipaddr:554/stream1

stream2.codec = h.264
stream2.size = 1280x720
stream2.address = rtsp://ipaddr:554/stream2

stream3.codec = mjpeg
stream3.size = 640x480
stream3.address = rtsp://ipaddr:554/stream3
```

<**If Error**>

```
Error:<description>
```

5. Event Request

5.1 Event

```
  http://<servername>/cgi-bin/control/requestevent.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getevent	get : Get curret value
eventtype=<string>	alarmin alarmout motion videotamper audiodetect all	alarmin/alarmout status, motion status [response style] alarmin# = on/off alarmout# = on/off motion# = on/off videotamper = on/off *if supported audiodetect_high = on/off *if supported audiodetect_low = on/off audiodetect_warning_high = on/off audiodetect_warning_low = on/off # is Numbr of each event Current, only support 1

5.1.1 Send --Method: GET

Example 1. Get Event message

```
http://192.168.1.30/cgi-bin/control/requestevent.cgi?id=admin&passwd=admin
&action=getevent&eventtype=all
```

5.1.2 Response

return :

HTTP Status: 200 OK

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

Body:

<**If Success**> -Where boundary is “**eventlist**”

```
--<boundary>
Content-type: text/plain₩n₩n
Content-Length: <data size (Byte)> ₩n
<data>
--<boundary>
```

<**If Error**>

```
Error:<description>
```

6. Alarm Out Control

6.1 Alarm Out(Relay) On/Off

```
 http://<servername>/cgi-bin/control/ctrl_alarmout.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	setalarmout	set: Set curret value
alarmout#=<string>	on off	[response style] alarmout# = on/off # is Number of each alarm-out Current, only support 1

6.1.1 Send --Method: GET

Example 1. Set

```
http://192.168.1.30/cgi-bin/control/ctrl_alarmout.cgi?id=admin&passwd=admin
&action=setalarmout&alarmout1=on
```

6.1.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
alarmout1 = on
```

<**If Error**>

```
Error:<description>
```

7. Image Transmission

7.1 JPEG Image Transmission

```
http://<servername>/cgi-bin/image/jpeg.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		

7.1.1 Send --Method: GET

Example 1. Get JPEG Stream

```
http://192.168.1.30/cgi-bin/image/jpeg.cgi?id=admin&passwd=admin
```

7.1.2 Response

return :

HTTP Status: 200 OK

<**If Success**>

```
Content-type : image/jpeg
Content-Length:<image size (byte)>
<JPEG Image Data>
```

<**If Error**>

```
Content-type : text/plain
Error:<description>
```

7.2 MJPEG Image Transmission

* IE(Microsoft) ,Opera and Android(Chrome) does not support this protocol.

```
http://<servername>/cgi-bin/image/mjpeg.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
source=<string>	input1 input2	Image Source
refresh=<int>	0~300	Fast (0) < -- > Slow(300)

7.2.1 Send --Method: GET

Example 1. Get MJPEG Stream

```
http://192.168.1.30/cgi-bin/image/mjpeg.cgi?id=admin&passwd=admin
```

7.2.2 Response

return :

HTTP Status: 200 OK

<**If Success**>

```
Content-type: multipart/x-mixed-replace;boundary=<boundary>
--<boundary>
Content-type: image/jpeg
Content-Length: <image size (Byte)>        <Repeated data>
<JPEG image data>
<Repeated data>
```

<**If Error**>

```
Content-type : text/plain
Error:<description>
```

7.2.3 Example

```
HTTP/1.1 200 OK
Content-type: multipart/x-mixed-replace;boundary=image_boundary_at
--image_boundary _at
Content-type: image/jpeg
Content-Length: <50603>
<JPEG image data>
--image_boundary_at
Content-type: image/jpeg
Content-Length: <48331>
<JPEG image data>
```

8. PTZ Control

* supported model only

8.1 Pan/Tilt/Zoom/Focus Status

```
http://<servername>/cgi-bin/control/ptzf_status.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getptzstatus setptzstatus lensreset	get ptz on/off status set ptz on/off status lens postion initialize
ptstatus=<string>	enable disable	set pan/tilt status
zfstatus=<string>	enable disable	set zoom/focus status
action=<string>	goptzfpos getptzfpos	go ptzf position Get ptzf position
panpos=<int>	0~35999	(0 ~ 359.99 deg.)x100
tiltpos=<int>	-2000~9000	(-20 ~ 90 deg.)x100
zoompos=<int>	0~65535	
focuspos=<int>	0~65535	supported model only
panspeed=<int>	1~100	1~100, 0: No action
tiltspeed=<int>	1~100	1~100, 0: No action
zoomspeed=<int>	1~100	1~100, 0: No action
focusspeed=<int>	1~100	1~100, 0: No action

8.1.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/ptzf_status.cgi?id=admin&passwd=admin
&action=getptzstatus
http://192.168.1.30/cgi-bin/control/ptzf_status.cgi?id=admin&passwd=admin
&action=setptzstatus&zfstatus=enable
http://192.168.1.30/cgi-bin/control/ptzf_status.cgi?id=admin&passwd=admin
&action=lensreset
```

Example 2.

```
http://192.168.1.30/cgi-bin/control/ptzf_status.cgi?id=admin&passwd=admin
&action=goptzfpos&panpos=9000&tiltpos=1000&panspeed=100&tiltspeed=100

http://192.168.1.30/cgi-bin/control/ptzf_status.cgi?id=admin&passwd=admin
&action=getptzfpos
```

8.1.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
ptstatus = disable
zfstatus = enable
```

<**If Error**>

```
Error:<description>
```

8.2 Pan/Tilt Control

```
http://<servername>/cgi-bin/control/pt_control.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	setptmove	set: pan/tilt movement
pan=<string>	right left stop	move direction
panspeed=<int>	1~100	
tilt=<string>	up down stop	move direction
tiltspeed=<int>	1~100	

8.2.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/pt_control.cgi?id=admin&passwd=admin
&action=setptmove&pan=right&panspeed=50&tilt=down&tiltspeed=50
```

8.2.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**> : **No reture Message**

```

```

<**If Error**>

```
Error:<description>
```

8.3 Zoom/Focus Control

```
http://<servername>/cgi-bin/control/zf_control.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	onepush setzfmove	one-push AF set zoom/focus
zoom=<string>	in out stop	move direction
zoomspeed=<int>	1~100	
focus=<string>	in out stop	move direction
focusspeed=<int>	1~100	

8.3.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/zf_control.cgi?id=admin&passwd=admin
&action=onepush
http://192.168.1.30/cgi-bin/control/zf_control.cgi?id=admin&passwd=admin
&action=setzfmove&zoom=in&zoomspeed=50
http://192.168.1.30/cgi-bin/control/zf_control.cgi?id=admin&passwd=admin
&action=setzfmove&focus=out&focusspeed=50
```

8.3.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**> : **No reture Message**

```
```

<**If Error**>

```
Error:<description>
```

8.4 Preset/ Auto pan

http://<*servername*>/cgi-bin/control/preset_control.cgi
[?<*argument*>=<*value*>[&<*argument*>=<*value*>...]]

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	setpreset gopreset clearpreset	preset
number=<int>	1~128,(255)	Max number of preset
action=<string>	autopan	auto pan
pos_a=<int>	1~128,(255)	move point A ←→B , see B3.2 Get Capabilities PTZ
pos_b=<int>	1~128,(255)	
speed=<int>	1~(100), 255:Stop	
action=<string>	autopan_cw autopan_ccw	auto pan if supported
speed=<int>	1~100, 255:stop	auto pan speed if supported

8.4.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/preset_control.cgi?id=admin&passwd=admin
&action=gopreset&number=1
http://192.168.1.30/cgi-bin/control/preset_control.cgi?id=admin&passwd=admin
&action=autopan&pos_a=1&pos_b=2&speed=30
http://192.168.1.30/cgi-bin/control/preset_control.cgi?id=admin&passwd=admin
&action=autopan_cw&speed=30
```

8.4.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**> : **No reture Message**

<**If Error**>

```
Error:<description>
```

8.5 PTZ Centering

```
  http://<servername>/cgi-bin/control/ptz_centering.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	setcenter	
type=<string>	box point	Max number of preset
center.startx=<int>	0~1920	This will be used when the type is **box**. center.startx ≤ center.endx center.starty ≤ center.endy * supported model only.
center.starty=<int>	0~1080	
center.endx=<int>	0~1920	
center.endy=<int>	0~1080	
center.pointx=<int>	0~1920	This will be used when the type is **point**. * supported model only.
center.pointy=<int>	0~1080	
speed=<int>	1~100	

8.5.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/ptz_centering.cgi?id=admin&passwd=admin
&action=setcenter&type=box&center.startx=300&center.starty=300&center.endx=500&cent
er.endy=500
http://192.168.1.30/cgi-bin/control/ptz_centering.cgi?id=admin&passwd=admin
&action=setcenter&type=point&center.pointx=300&center.pointy=300
```

8.5.2 Response

return :

HTTP Status:

Content-type : text/plain

Body:

<**If Success**> : **No reture Message**

<**If Error**>

```
Error:<description>
```

Annex A. Unified Command 1

```
http://<servername>/cgi-bin/control/serverinfo1.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getsysinfo1	get: system information

A1.1 Send --Method: GET

Example 1. Set

```
http://192.168.1.30/cgi-bin/control/serverinfo1.cgi?id=admin&passwd=admin
&action=getsysinfo1
```

A1.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[ServerName]
servername = welcome

[ServerDate]
serverdate = <month> <day>, <year> <hour>:<minute>:<second>

[ServerMac]
mac = xx:xx:xx:xx:xx:xx

[ColorInfo]
bright = 52
contrast = 5                * if supported model
saturation = 100
sharp = 100
edge = 68                   * if supported model
hue = 68                    * if supported model
```

```
[VideoInfo]
videoflip = normal
capture1.enable = enable
capture1.size = 1280x960
capture2.enable = enable
capture2.size = 640x480
encoder1.enable = enable
encoder1.source = capture1
encoder1.codec = h264
encoder1.framerate = 30
encoder1.quality = 512
encoder1.keyframe = 30
encoder1.ratectrl = vbr
encoder2.enable = enable
encoder2.source = capture2
encoder2.codec = mpeg
encoder2.framerate = 30
encoder2.quality = 512
encoder2.keyframe = 30
encoder2.ratectrl = vbr
encoder3.enable = enable
encoder3.source = capture1
encoder3.codec = jpeg
encoder3.framerate = 30
encoder3.quality = 50

[AudioInfo]
audiocodec = ulaw
audioout = disable
audioin = enable

[RTSPInfo]
rtspport = 554
rtpport = 5000,5999
rtcpenable = enable
rtsptimelimit = 0
multicastenable = disable
multicastttl = 5
multicastvidioip = 239.249.119.123
```

```
multicastvidioport = 40000
multicastaudioip = 239.249.119.129
multicastaudioport = 4002
authorityenable = enable
```

Annex A. Unified Command 2

```
http://<servername>/cgi-bin/control/serverinfo2.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getsysinfo2	get: system information

A2.1 Send --Method: GET

Example 1. Set

```
http://192.168.1.30/cgi-bin/control/serverinfo2.cgi?id=admin&passwd=admin
&action=getsysinfo2
```

A2.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[AlarmMotion]
allstatus = enable
maxno = 1

motion1.enable = enable
motion1.name = md1
motion1.level = 5
motion1.area1 = 15
```

```
motion1.area2 = 15
motion1.area3 = 15
motion1.area4 = 255
motion1.area5 = 255
motion1.area6 = 255

[AlarmIput]
allstatus = enable
maxno = 1

alarmin1.enable = enable
alarmin1.name = sensor
alarmin1.type = no

[AlarmOutput]
allstatus = enable
maxno = 1

alarmout1.enable = enable
alarmout1.name = relay
alarmout1.link = 3
alarmout1.time = 5

[EventRecord]
record.status = enable
record.streamid = stream1
record.link = 3
record.save = 1
record.timeprev = 5
record.timenext = 20
record.maxsize = 2048
```

<**If Error**>

```
Error:<description>
```

Annex A. Unified Command 3

```
 http://<servername>/cgi-bin/control/serverinfo3.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getsysinfo3	get: system information

A3.1 Send --Method: GET

Example 1. Set

```
http://192.168.1.30/cgi-bin/control/serverinfo3.cgi?id=admin&passwd=admin
&action=getsysinfo3
```

A3.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[CamDNF]
daynightmode = day
autolevel = 50
irlink = enable
lprdayspeed = 10       * if daynightmode is lpr
lprnightspeed = 7

[CamColor]
bright = 52
contrast = 5
saturation = 100
sharp = 100
edge = 68

[CamWhiteBalance]
```

```
wbmode = user
userred = 50
userblue = 50

[CamWDR]
wdrstatus = enable
mode = dwdr
dwdrmode = step1

[CamEffect]
colorbar = disable
monoimg = disable
negative = disable

[CamSlowShutter]
slowshutter = 25

[CamShutterSpeed]
shutmode = user
shutspeed = 9
agcvalue = 60
minshutspd = 6          * if shutmode is set auto, it will be display.
```

<**If Error**>

```
Error:<description>
```

Annex B. Get Capabilities

B1.1 Get Capabilities Video All

```
 http://<servername>/cgi-bin/control/capabilityvideo.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getCapabilitiesVideoAll	get : Get Capability

B1.1.1 Send --Method: GET

Example 1. if camera is 3M.

```
http://192.168.1.30/cgi-bin/control/capabilityvideo.cgi?id=admin&passwd=admin
&action=getCapabilitiesVideoAll
```

B1.1.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

[**Example for 2M Camera**]

```
[Capabilities Video]
MaxVideosizeH265 = 1920x1080
MaxVideosizeH264 = 1920x1080
MaxVideosizeMJPEG = 1920x1080
MaxVideosizeMPEG4 = 0x0

[Capabilities Video Stream]
StreamMethod = RTPUDP;RTPTCP;RTPHTTP
MaxStreamSupported = 3
stream1 = H265@Capture1;H264@Capture1
stream2 = H265@Capture2;H264@Capture2;MJPEG@Capture2
stream3 = H265@Capture3;H264@Capture3;MJPEG@Capture3

EventStreamSupport = OnvifMetadata

[Capabilities VideoCodec]
MaxVideoCodecsize = 3
VideoCodecSupported = H.265;H.264;MJPEG
```

```
CodecH265Maxuse = 3
CodecH264Maxuse = 3
CodecMPEG4Maxuse = 0
CodecMJPEGMaxuse = 2

[Capabilities Resolution]
VideoSizeAll = 1920x1080;1280x960;1280x720;1024x768;854x480;640x480;640x360;320x240;320x180
CaptureMaxsize = 3
Capture1 = 1920x1080;1280x960;1280x720;1024x768
Capture2 = 1920x1080;1280x960;1280x720;1024x768;854x480;640x480;640x360;320x240;320x180
Capture3 = 1280x720;1024x768;854x480;640x480;640x360;320x240;320x180

[Capabilities Framerate]
MaxFramerate = 30

min:1fps;max:30fps@VideoSizeAll

[Capabilities KeyFrame]
min:5;max:60@min:1fps;max:30fps

[Capabilities Bitrate]
BitrateAll = min:128kbps;max:8192kbps
JPEGBitrateSupported = Not_Support
MPEG4BitrateSupported = Not_Support
H.264BitrateSupported = BitrateAll
H.265BitrateSupported = BitrateAll

[Capabilities Quality]
JPEGQualitySupported = min:5;max:90
MPEG4QualitySupported = Not_Support
H.264QualitySupported = Not_Support
H.265QualitySupported = Not_Support
```

[**Example for 3M Camera**]

```
[Capabilities Video]
MaxVideosizeH265 = 2048x1536
MaxVideosizeH264 = 2048x1536
MaxVideosizeMJPEG = 2048x1536
MaxVideosizeMPEG4 = 0x0

[Capabilities Video Stream]
StreamMethod = RTPUDP;RTPTCP;RTPHTTP
MaxStreamSupported = 3
stream1 = H265@Capture1;H264@Capture1
stream2 = H265@Capture2;H264@Capture2;MJPEG@Capture2
stream3 = H265@Capture3;H264@Capture3;MJPEG@Capture3

EventStreamSupport = OnvifMetadata

[Capabilities VideoCodec]
MaxVideoCodecsize = 3
VideoCodecSupported = H.265;H.264;MJPEG

CodecH265Maxuse = 3
CodecH264Maxuse = 3
CodecMPEG4Maxuse = 0
```

```
CodecMJPEGMaxuse = 2

[Capabilities Resolution]
VideoSizeAll = 2048x1536;1600x1200;1920x1080;1280x960;1280x720;1024x768;854x480;640x480;640x360;320x240;320x180
CaptureMaxsize = 3
Capture1 = 2048x1536;1600x1200;1920x1080;1280x960;1280x720;1024x768
Capture2 = 2048x1536;1600x1200;1920x1080;1280x960;1280x720;1024x768;854x480;640x480;640x360;320x240;320x180
Capture3 = 640x480;640x360;320x240;320x180

[Capabilities Framerate]
MaxFramerate = 30

min:1fps;max:30fps@VideoSizeAll

[Capabilities KeyFrame]
min:5;max:60@min:1fps;max:30fps

[Capabilities Bitrate]
BitrateAll = min:128kbps;max:8192kbps
JPEGBitrateSupported = Not_Support
MPEG4BitrateSupported = Not_Support
H.264BitrateSupported = BitrateAll
H.265BitrateSupported = BitrateAll

[Capabilities Quality]
JPEGQualitySupported = min:5;max:90
MPEG4QualitySupported = Not_Support
H.264QualitySupported = Not_Support
```

B1.2 Get Capabilities Video

```
http://<servername>/cgi-bin/control/capabilityvideo.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getVideo	get : Get value

<Response>

MaxVideosizeH265 = <string>	12M: 4000x3000	Supported Max Resolutions *H.256:supported model only
MaxVideosizeH264 = <string>	9M: 3000x3000	
MaxVideosizeMJPEG = <string>	8M: 3840x2160	
MaxVideosizeMPEG4 = <string>	5M: 2560x1920 4M: 2560x1440 3M: 2048x1536 2M: 1920x1080 1M: 1280x1024	
StreamMethod = <string>	RTPUDP;RTPTCP;RTPHTTP	
MaxStreamSupported = <int>		
stream#1 = Codec@<string>	#1: 1<= # <= MaxStreamSupported Capture#2 Each Resolutions	Codec: H.264,MPEG4,MJPEG Capture#2 : see Resolution
EventStreamSupport = <string>	No OnvifMetadata UserFormat	* onvif Metadata format of RTSP

B1.2.1 Send --Method: GET

Example 1. if camera is 3M.

```
http://192.168.1.30/cgi-bin/control/capabilityvideo.cgi?id=admin&passwd=admin
&action=getVideo
```

B1.2.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities Video]
MaxVideosizeH265 = 2048x1536
MaxVideosizeH264 = 2048x1536
MaxVideosizeMJPEG = 2048x1536
MaxVideosizeMPEG4 = 0x0

[Capabilities Video Stream]
StreamMethod = RTPUDP;RTPTCP;RTPHTTP
MaxStreamSupported = 3
stream1 = H265@Capture1;H264@Capture1
stream2 = H265@Capture2;H264@Capture2;MJPEG@Capture2
stream3 = H265@Capture3;H264@Capture3;MJPEG@Capture3

EventStreamSupport = OnvifMetadata
```

B1.3 Get Capabilities VideoCodec

```
 http://<servername>/cgi-bin/control/capabilityvideo.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getVideoCodec	get : Get value

<Response>

MaxVideoCodec = <int>		
VideoCodecSupported = <string>	H.265 H.264 MPEG4 MJPEG	Supported Codecs *H.256:supported model only
CodecH264Maxuse = <int>	0 ~	Max usable counter of codec.
CodecH265Maxuse = <int>		
CodecMPEG4Maxuse = <int>	0 ~	
CodecMJPEGMaxuse = <int>	0 ~	

B1.3.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/capabilityvideo.cgi?id=admin&passwd=admin
&action=getVideoCodec
```

B1.3.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities VideoCodec]
MaxVideoCodecsize = 3
VideoCodecSupported = H.265;H.264;MJPEG

CodecH265Maxuse = 3
CodecH264Maxuse = 3
CodecMPEG4Maxuse = 0
CodecMJPEGMaxuse = 2
```

B1.4 Get Capabilities Resolution

```
http://<servername>/cgi-bin/control/capabilityvideo.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getResolution	get : Get value

<Response>

VideoSizeAll = <string>	3M: 2048x1536; ~ ;320x240 2M: 1920x1080; ~ ;320x240 1M: 1280x1024; ~ ;320x240	
MaxCapture = <int>	1~30	the number of Capture#
Capture# = <string>	VideoSizeAll Each Resolutions	# is capture number.

B1.4.1 Send --Method: GET

Example 1. if camera is 3M.

```
http://192.168.1.30/cgi-bin/control/capabilityvideo.cgi?id=admin&passwd=admin
&action=getResolution
```

B1.4.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities Resolution]
VideoSizeAll = 2048x1536;1600x1200;1920x1080;1280x960;1280x720;1024x768;854x480;640x480;640x360;320x240;320x180
CaptureMaxsize = 3
Capture1 = 2048x1536;1600x1200;1920x1080;1280x960;1280x720;1024x768
Capture2 = 2048x1536;1600x1200;1920x1080;1280x960;1280x720;1024x768;854x480;640x480;640x360;320x240;320x180
Capture3 = 640x480;640x360;320x240;320x180
```

B1.5 Get Capabilities Framerate & Keyframe

http://<*servername*>/cgi-bin/control/capabilityvideo.cgi
[?<*argument*>=<*value*>[&<*argument*>=<*value*>...]]

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getFramerate	get : Get value

<Response>

MaxFramerate = <int>	1~	
fix:#fps@resolutions	#is 60,45, etc	Fixed frame @ resoutions
min:#1fps;max:#2fps@resolutions	#1: 1~ #2: 1~	# 1,#2 : integer value
min;#1;max:#2@fps	#1,#2: integer	Keyframe value for the Framerate.

B1.5.1 Send --Method: GET

Example 1. if camera is 1M.

```
http://192.168.1.30/cgi-bin/control/capabilityvideo.cgi?id=admin&passwd=admin
&action=getFramerate
```

B1.5.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities Framerate]
MaxFramerate = 30

min:1fps;max:30fps@VideoSizeAll

[Capabilities KeyFrame]
min:5;max:60@min:1fps;max:30fps
```

B1.6 Get Capabilities Bitrate

```
http://<servername>/cgi-bin/control/capabilityvideo.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getBitrate	get : Get value

<Response>

BitrateAll = min:#1kbps;max:#2kbps	#1: 64 #2: integer over 64	
JPEGBitrateSupported = <string>	Not_Support BitrateAll min:#1kbps;max:#2kbps	# 1,#2 : integer
MPEG4BitrateSupported = <string>		
H.264BitrateSupported = <string>		
H.265BitrateSupported = <string>		*H.256:supported model only

B1.6.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/capabilityvideo.cgi?id=admin&passwd=admin
&action=getBitrate
```

B1.6.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities Bitrate]
BitrateAll = min:128kbps;max:8192kbps
JPEGBitrateSupported = Not_Support
MPEG4BitrateSupported = Not_Support
H.264BitrateSupported = BitrateAll
H.265BitrateSupported = BitrateAll
```

B1.7 Get Capabilities Quality

```
http://<servername>/cgi-bin/control/capabilityvideo.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getQuality	get : Get value

<Response>

JPEGQualitySupported = <string>	Not_Support min:#1;max:#2	#1,#2 : integer value
MPEG4QualitySupported = <string>		
H.264QualitySupported = <string>		
H.265QualitySupported = <string>		*H.256:supported model only

B1.7.1 Send --Method: GET

Example 1. if camera is 3M.

```
http://192.168.1.30/cgi-bin/control/capabilityvideo.cgi?id=admin&passwd=admin
&action=getQuality
```

B1.7.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities Quality]
JPEGQualitySupported = min:5;max:90
MPEG4QualitySupported = Not_Support
H.264QualitySupported = Not_Support
H.265QualitySupported = Not_Support
```

B2.1 Get Capabilities All Audio

```
 http://<servername>/cgi-bin/control/capabilityaudio.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getCapabilitiesAudioAll	get : Get Capability

B2.1.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/capabilityaudio.cgi?id=admin&passwd=admin
&action=getCapabilitiesAudioAll
```

B2.1.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities Audio]
AudioInputSupported = Yes
AudioOutputSupported = Yes

AudioUploadStream = RTSP
AudioDownloadStream = RTSP

[Capabilities AudioCodec]
MaxAudioInputChannel = 1
AudioInputCodecSupported = G.711

MaxAudioOutputChannel = 1
AudioOutputCodecSupported = G.711

[CodecInfo G.711]
SamplingAll = 8K
SamplingFreq = min:8K;max:8K
Compress = ulaw
```

B2.2 Get Capabilities Audio

```
 http://<servername>/cgi-bin/control/capabilityaudio.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getAudio	get : Get value

<Response>

AudioInputSupported = <string>	Yes No	
AudioOutputSupported = <string>	Yes No	
AudioUploadStream = <string>	RTSP CGI No	Now, Support RTSP only * Camera → Client
AudioDownloadStream = <string>	RTSP CGI No	Now, Support RTSP only * Clint → Camera

B2.2.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/capabilityaudio.cgi?id=admin&passwd=admin
&action=getAudio
```

B2.2.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities Audio]
AudioInputSupported = Yes
AudioOutputSupported = Yes

AudioUploadStream = RTSP
AudioDownloadStream = RTSP
```

B2.3 Get Capabilities AudioCodec

```
 http://<servername>/cgi-bin/control/capabilityaudio.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getAudioCodec	get : Get value

<Response>

MaxAudioInputChannel = <int>	0~	
AudioInputCodecSupported = <string>	G.711 G.723 AAC No	Now, G.711 Only
MaxAudioOutputChannel = <int>	0~	
AudioOutputCodecSupported = <string>	G.711 G.723 AAC No	Now, G.711 Only

B2.3.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/capabilityaudio.cgi?id=admin&passwd=admin
&action=getAudioCodec
```

B2.3.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities AudioCodec]
MaxAudioInputChannel = 1
AudioInputCodecSupported = G.711

MaxAudioOutputChannel = 1
AudioOutputCodecSupported = G.711

[CodecInfo G.711]
```

```
SamplingAll = 8K
SamplingFreq = min:8K;max:8K
Compress = ulaw
```

B3.1 Get Capabilities All PTZ

```
http://<servername>/cgi-bin/control/capabilityptz.cgi
[?<argument>=<value>[&<argument>=<value>...]]
```

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getCapabilitiesPTZAll	get : Get Capability

B3.1.1 Send --Method: GET

Example 1.

```
http://192.168.1.30/cgi-bin/control/capabilityptz.cgi?id=admin&passwd=admin
&action=getCapabilitiesPTZAll
```

B3.1.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities PTZ]
PanSupported = No
TiltSupported = No
ZoomSupported = Yes
FocusSupported = Yes
AutopanSupported = No
PresetSupported = No
```

B3.2 Get Capabilities PTZ

http://<*servername*>/cgi-bin/control/capabilityptz.cgi
[?<*argument*>=<*value*>[&<*argument*>=<*value*>...]]

Argument	Values	Description
id=<string>		
passwd=<string>		
action=<string>	getPTZ	get : Get value

<Response>

PanSupported = <string>	Yes No	
TiltSupported = <string>	Yes No	
ZoomSupported = <string>	Yes No	
FocusSupported = <string>	Yes No	
EndlessPanSupported = <string>	Yes No	
AutoFocusSupported = <string>	Yes No	
PresetSupported = <string / int>	No 1~256	Max Preset Number
AutopanSupported = <string / int>	No 1~100	Max Autopan Number
AutopancwSupported = <string>	Yes No	Auto pan cw/ccw support
TourSupported = <string / int>	No 1~100	Max Tour Number

B3.2.1 Send --Method: GET

Example 1.

http://192.168.1.30/cgi-bin/control/capabilityptz.cgi?id=admin&passwd=admin
&action=getPTZ

B3.2.2 Response

return :

HTTP Status: 200 OK

Content-type : text/plain

Body:

<**If Success**>

```
[Capabilities PTZ]
PanSupported = No
TiltSupported = No
ZoomSupported = Yes
FocusSupported = Yes

EndlessPanSupported = No
AutoFocusSupported = No

PresetSupported = No
AutopanSupported = No
AutopancwSupported = No
TourSupported = No
```