IP Camera Open API, HTTP - Interface Specification

DOCUMENT HISTORY

Version	Date	Supported Firmware	Release Notes
1.00	2007-May-14	V3.0.0.0	Initial version
1.06	2007-Dec-28	V3.0.2.1691	Revise Document
2.00	2008-May-22	V5.0.0.0	h264 version
3.00	2009-Dec-10	V5.0.1.3714	API re-define.
3.01	2010-Jan-07	V5.0.2.3843	Revise Document

---

1 General

1.1 Add, update, remove and list

1.1.1 List parameters

Syntax:

```
http://<cameraname>/config.cgi?action=list

[&<parameter>=<value>...]
```

1.1.2 List output format

```
HTTP/1.0 200 OK\r\n

Content-Type: text/plain\n

\n

<parameter pair>

where <parameter pair> is

<parameter>=<value>\n

[ <parameter pair> ]
```

1.1.3 Update parameters

Syntax:

```
http://<cameraname>/config.cgi?action=update[&<parameter>=<value>...]
```

1.1.4 Add parameters

Syntax:

```
http://<cameraname>/config.cgi?action=add[&<parameter>=<value>...]
```

1.1.5 Remove parameters

Syntax:

```
http://<cameraname>/config.cgi?action=remove[&<parameter>=<value>...]
```

1.1.6 Add/Remove server responses

Return: A successful add.

```
HTTP/1.0 200 OK\r\n

Content-Type: text/plain\n

\n

<entry> OK\r\n
```

Return: A successful remove.

```
HTTP/1.0 200 OK\r\n

Content-Type: text/plain\n

\n

OK\r\n
```

1.2 Add, modify and delete users

Syntax:

```
http://<cameraname>/usrgrp.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

1.3 Factory default

Reload factory default. All parameters except Network.BootProto, Network.IPAddress, Network.SubnetMask, Network.Broadcast and Network.DefaultRouter are set to their factory default values.

Syntax:

```
http://<cameraname>/factorydefault.cgi
```

1.4 Hard factory default

All parameters are set to their factory default value.

Syntax:

```
http://<cameraname>/hardfactorydefault.cgi
```

1.5 Backup

Download backup.bin into PC.

Syntax:

```
http://<cameraname>/backup.cgi
```

Return:

```
HTTP/1.0 200 OK\r\n
Content-Type: application/octet-stream\r\n
Content-Disposition: attachment; filename=backup.bin\r\n
\r\n
<file content of backup.bin>
```

1.6 Restore

Upload a unit specific backup previously created by the backup.cgi.

Syntax:

```
http://<cameraname>/restore.cgi
```

1.7 Firmware upgrade

Upgrade the firmware version.

Syntax:

```
http://<cameraname>/firmwareupgrade.cgi
```

1.8 Reboot server

Restart server.

Syntax:

```
http://<cameraname>/reboot.cgi
```

1.9 System logs

Syntax:

```
http://<cameraname>/systemlog.cgi
```

Return:

```
HTTP/1.0 200 OK\r\n
Content-Type: text/plain\r\n
\r\n
<system log information>
```

1.10 System date and time

Syntax:

```
http://<cameraname>/time.cgi?<parameter>=<value>
```

1.10.1 Get system date and time

Syntax:

```
http://<cameraname>/time.cgi?action=get
```

Return:

```
HTTP/1.0 200 OK\r\n

Content-Type: text/plain\r\n

\r\n

<month> <day>, <year> <hour>:<minute>:<second>\r\n
```

1.10.2 Set system date and time

Syntax:

```
http://<cameraname>/time.cgi?action=set[&<parameter>=<value>...]
```

The set action produces one of the following server responses:

Return: A successful set.

```
HTTP/1.0 200 OK\r\n

Content-Type: text/plain\r\n

\r\n

OK\r\n
```

1.11 OSD string

Syntax:

```
http://<cameraname>/setosd.cgi?<parameter>=<value>
```

2 Image and Video

2.1 Image size

Syntax:

```
http://<cameraname>/resolution.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

2.2 Video status

Syntax:

```
http://<cameraname>/videostatus.cgi?<parameter>=<value>
```

2.3 JPEG/MJPG

2.3.1 JPEG image request

Syntax:

```
http://<cameraname>/image.jpg
```

2.3.2 JPEG image (snapshot) CGI request

Syntax:

```
http://<cameraname>/image.cgi[?<parameter>=<value>[&<parameter>=<value>...]]
```

2.3.3 JPEG image response

Return:

```
HTTP/1.0 200 OK\r\n

Content-Type: image/jpeg\r\n

Content-Length: <image size>\r\n

\r\n
```

```
<JPEG image data>\r\n
```

2.3.4 MJPG video request

Syntax: Request Multipart JPEG image.

```
http://<cameraname>/video.mjpg
```

2.3.5 MJPG video CGI request

Syntax:

```
http://<cameraname>/video.cgi

[?<parameter>=<value>[&<parameter>=<value>...]]
```

2.3.6 MJPG video response

Return:

```
HTTP/1.0 200 OK\r\n

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

\r\n

--<boundary>\r\n

<image>

where the proposed <boundary> is

myboundary

and the returned <image> field is

Content-Type: image/jpeg\r\n

Content-Length: <image size>\r\n

\r\n

<JPEG image data>\r\n
```

```
--<boundary>\r\n

<image>
```

3 PTZ

3.1 PTZ control

Syntax:

```
http://<cameraname>/ptz.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

3.2 PTZ configuration and OSD control

Syntax:

```
http://<cameraname>/configptz.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

4 Motion Detection

4.1 Add a Motion Detection window

Example: Add a new Motion Detection window with default values.

```
http://camserver/config.cgi?action=add&group=Motion&template=motion
```

Example: Add a new Motion Detection window with specified values.

```
http://camserver/config.cgi?action=add&group=Motion&template=motion&Motion.M.Name=Entrance&Motion.M.Top=500&

Motion.M.Bottom=7000&Motion.M.Left=5000&Motion.M.Right=8500
```

Return:

```
HTTP/1.0 200 OK\r\n

Content-Type: text/plain\r\n

\r\n

M<group number> OK\r\n
```

4.2 Remove a Motion Detection window

Example: Remove Motion Detection window defined within Motion.M3 and Motion.M5.

```
http://camserver/config.cgi?action=remove&group=Motion.M3,group=Motion.M5
```

4.3 Update the Motion Detection parameters

Example: Update the parameters for an existing Motion Detection window.

```
http://camserver/config.cgi?action=update&Motion.M1.Top=1500
```

```
&Motion.M1.Bottom=8000
```

4.4 List the Motion detection parameters

Example: List the Motion.M1 and Motion.M2 parameters.

```
http://camserver/config.cgi?action=list&group=Motion.M1,group=Motion.M2
```

Example: List all Motion Detection windows.

```
http://camserver/config.cgi?action=list&group=Motion
```

4.5 Get the Motion Detection level

Syntax:

```
http://<cameraname>/motion.cgi[?<parameter>=<value>...]
```

Return:

```
HTTP/1.0 200 OK\r\n
Content-Type: multipart/x-mixed-replace;boundary=<boundary>\r\n
\r\n
--<boundary>\r\n
<motion levels>
where the proposed boundary <boundary> is mdb
and the <motion levels> part is
Content-Type: text/plain\r\n
\r\n
<motion level for window with lowest group number>
--<boundary>\r\n
```

and <motion level for window with group number n>" is

group=<group number n>;level=<motion level for n>;threshold=

<threshold level for n>;\r\n[<motion level for window n+1>]

5 I/O

5.1 I/O control

Syntax:

```
http://<cameraname>/port.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

5.2 Input

Syntax:

```
http://<cameraname>/input.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

Return: "monitor", i.e., multipart "check" parameter

```
HTTP/1.0 200 OK\r\n

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

\r\n

--<boundary>\r\n

<monitor data>

where the proposed boundary <boundary> is

ioboundary

and the <monitor data> part is

Content-Type: text/plain\r\n

\r\n

<check data>

--<boundary>\r\n

and <check data> is
```

IO<n>:<char>\r\n

and <n> is the I/O port number and <char> is / or H when the port is active and \ or L when the port is inactive.

Note: The output can contain extra blank lines, i.e., extra \r\n within the sections.

5.3 Output

Syntax:

http://<cameraname>/output.cgi?<parameter>=<value>[&<parameter>=<value>...]

6 IP filter

6.1 IP filter administration

Syntax:

```
http://<cameraname>/ipfilter.cgi?<parameter>=<value>[&<parameter>=<value>... ]
```

6.2 Server responses

Return: A successful add, remove, removeall or update.

```
HTTP/1.0 200 OK\r\n

Content-Type: text/plain\r\n

\r\n

OK\r\n
```

Return: A list.

```
HTTP/1.0 200 OK\r\n

Content-Type: text/plain\r\n

\r\n

Accept addresses: <IP addresses>\r\n

Enabled: <yes/no>\r\n
```

7 Audio

7.1 Audio MIME types

Supported MIME types for audio

7.2 Audio data request

Syntax:

```
http://<cameraname>/fromserver.cgi[?<parameter>=<value>]
```

7.3 Singlepart audio data response

Return:

```
HTTP/1.0 200 OK\r\n

Content-Type: <audio MIME>\r\n

\r\n

<Audio data>
```

7.4 Multipart audio data response

Return:

```
HTTP/1.0 200 OK\r\n

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

\r\n

--<boundary>\r\n

<audio>

where the proposed <boundary> is:

myboundary
```

```
and the <audio> field is

Content-Type: <audio MIME>\r\n

\r\n

<Audio data>\r\n

--<boundary>\r\n

<audio>
```

7.5 Audio data transmit

Syntax:

```
http://<cameraname>/toserver.cgi
```

There are no valid parameters and values.

8 Storage

The requests specified in the storage section are supported by products that have micro SD socket.

8.1 mount storage

mount / umount the storage

Method: GET/POST

Syntax:

```
http://<cameraname>/mount.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

with the following parameter and values

<parameter>=<value>	Values	Description
diskid=<string>	SD_DISK	specified the storage device
action=<string>	mount, umount	specified the action to mount/umount storage device

8.2 format storage

format the storage

Syntax:

```
http://<cameraname>/format.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

with the following parameter and values

<parameter>=<value>	Values	Description
diskid=<string>	SD_DISK	specified the storage device

8.3 list storage status

list the storage status

Syntax:

```
http://<cameraname>/list.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

with the following parameter and values

<parameter>=<value>	Values	Description
diskid=<string>	SD_DISK, all	specified the storage device

return value

```
<?xml version="1.0" ?>
  <root>
  <disks numberofdisks="1">
  <disk diskid="SD_DISK" totalsize="124048" freesize="124032" cleanuplevel="95" cleanupmaxage="1" cleanuppolicy="none"
    locked="no" full="no" readonly="no" status="OK" />
  </disks>
  </root>
```

8.4 list recording files

list the recording files

Syntax:

```
http://<cameraname>/list.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

with the following parameter and values

<parameter>=<value>	Values	Description
recordingid=<string>	all	specified the recording file

return value

```
<?xml version="1.0" ?>
<root>
<recordings>
<recording recordingid="20090101_010101.avi" diskid="SD_DISK" starttime="20090101_010101" date="20090101" recordingtype="scheduled">
</recording>
<recording recordingid="20090101_010201.avi" diskid="SD_DISK" starttime="20090101_010201" date="20090101" recordingtype="scheduled">
</recording>
</recordings>
```

8.5 remove recording file

remove the recording file

Syntax:

```
http://<cameraname>/remove.cgi?<parameter>=<value>[&<parameter>=<value>...]
```

with the following parameter and values

<parameter>=<value>	Values	Description
recordingid=<string>	all	specified the recording file