Blackmagicdesign

Blackmagic
Videohub
Ethernet Protocol

Blackmagic Videohub 12G
Universal Videohub
Smart Videohub CleanSwitch 12x12

Videohub Master Control Pro
Videohub Smart Control Pro
Blackmagic GPI and Tally Interface

Developer Information

Blackmagic Videohub Ethernet Protocol v2.3

Summary

The Blackmagic Videohub Ethernet Protocol is a text based protocol that is accessed by connecting to TCP port 9990 on a Videohub Server. Integrated Videohub Servers and Videohub Server computers are supported by the protocol.

The Videohub Server sends information in blocks which each have an identifying header in all-caps, followed by a full-colon. A block spans multiple lines and is terminated by a blank line. Each line in the protocol is terminated by a newline character.

Upon connection, the Videohub Server sends a complete dump of the state of the device. After the initial status dump, status updates are sent every time the Videohub status changes.

To be resilient to future protocol changes, clients should ignore blocks they do not recognize, up to the trailing blank line. Within existing blocks, clients should ignore lines they do not recognize.

LEGEND

↵   line feed or carriage return

•••   and so on

Version 2.3 of the Blackmagic Videohub Ethernet Protocol was released with Videohub 4.9.1 software.

Protocol Preamble

The first block sent by the Videohub Server is always the protocol preamble:

```
PROTOCOL PREAMBLE:↵

Version: 2.3↵

↵
```

The version field indicates the protocol version. When the protocol is changed in a compatible way, the minor version number will be updated. If incompatible changes are made, the major version number will be updated.

Device Information

The next block contains general information about the connected Videohub device. If a device is connected, the Videohub Server will report the attributes of the Videohub:

```
VIDEOHUB DEVICE:↵

Device present: true↵

Model name: Blackmagic Smart Videohub↵

Video inputs: 16↵

Video processing units: 0↵

Video outputs: 16↵

Video monitoring outputs: 0↵

Serial ports: 0↵

↵
```

This example is for the Smart Videohub which is a 16x16 router.

If the Videohub Server has no device connected, the block will simply be:

```
VIDEOHUB DEVICE:↵
Device present: false↵
↵
```

If a device is present, but has an incompatible firmware, the status reported will be:

```
VIDEOHUB DEVICE:↵
Device present: needs _ update↵
↵
```

In the last two situations, no further information will be sent, unless the situation is rectified.

If the Videohub Server detects a new Videohub attached, it will resend all blocks except the protocol preamble to indicate the device has changed, and allow the client to update its cache of server state.

Initial Status Dump

The next four blocks enumerate the labels assigned to the input, output, monitoring and serial ports.

Videohubs that do not have monitoring or serial ports do not send the corresponding blocks.

```
INPUT LABELS:↵
0 VTR 1↵
1 VTR 2↵
…
↵
OUTPUT LABELS:↵
0 Output feed 1↵
1 Output feed 2↵
…
↵
MONITORING OUTPUT LABELS:↵
0 Monitor feed 1↵
1 Monitor feed 2↵
…
↵
SERIAL PORT LABELS:↵
0 Deck 1↵
1 Deck 2↵
↵
```

NOTE Ports are always numbered starting at zero in the protocol which matches port one on the chassis.

The next three blocks describe the routing of the output, monitoring and serial ports.

```
VIDEO OUTPUT ROUTING:↵

0 5↵

1 3↵

…

↵

VIDEO MONITORING OUTPUT ROUTING:↵

0 7↵

1 8↵

…

↵

SERIAL PORT ROUTING:↵

0 12↵

1 11↵

…

↵
```

Videohubs with processing units (only the Workgroup Videohub) send an extra routing block:

```
PROCESSING UNIT ROUTING:↵

0 5↵

1 3↵

…

↵
```

Videohubs with frame buffers (only the Workgroup Videohub) send two extra blocks:

```
FRAME LABELS:↵

0 Frame one↵

1 Frame two↵

…

↵

FRAME BUFFER ROUTING:↵

0 7↵

1 8↵

…

↵
```

The next three blocks describe the locking status of the output, monitoring and serial ports. Each port has a lock status of "O" for ports that are owned by the current client (i.e., locked from the same IP address), "L" for ports that are locked from a different client, or "U" for unlocked. Note that Videohubs that do not have monitoring ports or serial ports do not send the corresponding blocks.

```
VIDEO OUTPUT LOCKS:↵

0 U↵
```

```
1 U↵
…
↵
MONITORING OUTPUT LOCKS:↵
0 U↵
1 U↵
…
↵
SERIAL PORT LOCKS:↵
0 U↵
1 U↵
…
↵
```

Videohubs with processing units (only the Workgroup Videohub) send an extra lock block:

```
PROCESSING UNIT LOCKS:↵
0 U↵
1 U↵
…
↵
```

Videohubs with frame buffers (only the Workgroup Videohub) send an extra lock block:

```
FRAME BUFFER LOCKS:↵
0 U↵
1 U↵
…
↵
```

Videohubs with serial ports next send a block which describes the direction of each serial port. Each port has a direction of either "control" for the "In (Workstation)" setting, "slave" for "Out (Deck)", or "auto" for "Automatic".

```
SERIAL PORT DIRECTIONS:↵
0 control↵
1 slave↵
2 auto↵
…
↵
```

Videohubs with pluggable cards (only Universal Videohubs) send three more blocks that describe the hardware status of the ports. Missing video or serial ports have a status of "None"; input and output video ports will be "BNC" or "Optical" if they are present; serial ports will be "RS422" if they are present.

```
VIDEO INPUT STATUS:↵
0 BNC↵
```

```
1 BNC↵
…↵
↵
VIDEO OUTPUT STATUS:↵
0 BNC↵
1 BNC↵
…↵
↵
SERIAL PORT STATUS:↵
0 RS422↵
1 RS422↵
…↵
↵
```

Status Updates

When any route, label, or lock is changed on the Videohub Server by any client, the Videohub Server resends the applicable status block, containing only the items that have changed. For example, if serial port 6 has been unlocked, the following block will be sent:

```
SERIAL PORT LOCKS:↵
5 U↵
↵
```

```
If multiple items are changed, multiple items may be present in
the update:
```
```
OUTPUT LABELS:↵
7 New output 8 label↵
10 New output 11 label↵
↵
```

If a card is plugged into or removed from the Universal Videohub, it will send hardware status blocks for the video inputs, video outputs, and serial ports on that card.

Requesting Changes

To update a label, lock or route, the client should send a block of the same form the Videohub Server sends when its status changes. For example, to change the route of output port 8 to input port 3, the client should send the following block:

```
VIDEO OUTPUT ROUTING:↵
7 2↵
↵
```

The block must be terminated by a blank line. On receipt of a blank line, the Videohub Server will either acknowledge the request by responding:

```
ACK↵
↵
```

or indicate that the request was not understood by responding:

```
NAK↵
↵
```

After a positive response, the client should expect to see a status update from the Videohub Server showing the status change. This is likely to be the same as the command that was sent, but if the request could not be performed, or other changes were made simultaneously by other clients, there may be more updates in the block, or more blocks. Simultaneous updates could cancel each other out, leading to a response that is different to that expected.

In the absence of simultaneous updates, the dialog expected for a simple label change is as follows:

```
OUTPUT LABELS:↵
6 new output label seven↵
↵
ACK↵
↵
OUTPUT LABELS:↵
6 new output label seven↵
↵
```

The asynchronous nature of the responses means that a client should never rely on the desired update actually occurring and must simply watch for status updates from the Videohub Server and use only these to update its local representation of the server state.

To lock a port, send an update to the port with the character "O" indicating that you wish to lock the port for example:

```
SERIAL PORT LOCKS:↵
7 O↵
↵
ACK↵
↵
SERIAL PORT LOCKS:↵
7 O↵
↵
```

To forcibly unlock a port that has been locked by another client, send an update to the port with the character "F" instead of using the usual unlock character "U". For example, to override a lock on port 7:

```
SERIAL PORT LOCKS:↵
7 F↵
↵
ACK↵
↵
SERIAL PORT LOCKS:↵
7 U↵
↵
```

Hardware status blocks can only be sent by the Videohub Server. If a client sends hardware status blocks, they will be ignored.

Requesting a Status Dump

The client may request that the Videohub Server resend the complete state of any status block by sending the header of the block, followed by a blank line. In the following example, the client requests the Videohub Server resend the output labels:

```
OUTPUT LABELS:↵
↵
ACK↵
↵
OUTPUT LABELS:↵
0 output label 1↵
1 output label 2↵
2 output label 3↵
…
↵
```

Checking the Connection

While the connection to the Videohub Server is established, a client may send a special no-operation command to check that the Videohub Server is still responding:

```
PING:↵
↵
```

If the Videohub Server is responding, it will respond with an ACK message as for any other recognized command.