Blackmagicdesign

Developer Information

Videohub

**Includes Blackmagic Videohub Ethernet Protocol
and Videohub RS-422 Protocol**

May 2018

Contents

Videohub

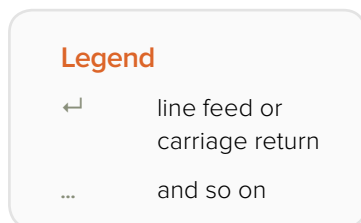# Blackmagic Videohub Ethernet Protocol v2.3

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.

Videohub RS-422 Protocol

General

The RS-422 protocol can be used to control many Videohub models as RS-422 slave devices, for router crosspoint switching. This feature is available on any Videohub model with an "RS-422 Control" port.

Depending on your model of Videohub, the RS-422 port may be either a DB9 or RJ11 connector. DB9 cables are commonly available in electrical stores. The RJ11 connector is the same used in many landline telephone connections. By modifying an RS-422 to USB adapter cable terminated with an RJ11 connector you can control Videohub using external controllers via USB.

In "Leitch Server" mode, these Videohub models implement the router (server) side of the Leitch Serial Pass-Through Protocol as specified in section 4 of Leitch document SPR-MAN revision D. In "Leitch Client" mode, the Videohub implements the controller (client) side of the Leitch terminal protocol.
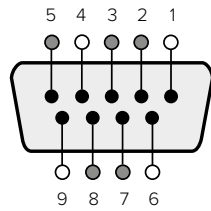
Pin	Function
1	Ground
2	Tx- (Data sent by router)
3	Rx+ (Data received by router)
4	Ground
5	NC (No connect)
6	Ground
7	Tx+ (Data sent by router)
8	Rx- (Data received by router)
9	Ground

Videohub RS-422 Router Control DB9 pin connections.

Pin No.	Function
Pin 1	TX +
Pin 2	TX -
Pin 3	GND
Pin 4	GND
Pin 5	RX -
Pin 6	RX +
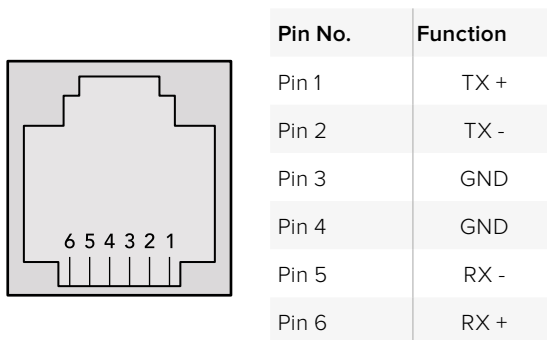
Videohub RS-422 Router Control RJ11 pinout diagram.

This document describes the commands and parameters in the protocol that are relevant and supported by Videohub. Other commands and parameters specified in the Leitch protocol are accepted but ignored.

The RS-422 serial port is configured as 9600 N81.

The protocol is line-oriented, with a maximum length of 250 characters per command. Each command from the client should be terminated with a carriage-return (\r). Each response from the server will be terminated with a carriage-return and line-feed (\r\n).

Sources, destinations and levels are always specified in base-16, numbered from zero. Levels are always between 0 and 15 ("F") . Videohubs only have one valid level - level zero.

On connecting to the serial port, the client should send a carriage-return. The Videohub Server will respond with a > character prompt, which is not followed by a carriage-return or line-feed. Receiving the prompt indicates that a connection has been established. The same prompt will be issued after each command received by the Videohub Server.

In the following documentation, commands in orange and values in blue must be typed literally, including any spaces. In the following example of an immediate command using destination port 7 and source port 3, @ X:0/destination,source would be entered as: @ X:0/6,2

Notifications

Once connected, if status reporting is enabled, the client will receive a notification message when a route changes on the Videohub Server. The notifications take one of two forms:

`S:0destination,source`   Routing change
This message indicates that the specified source port is now routed to the specified destination.

`V:0destination,source`   Preset routing notification

This message indicates that the current preset includes a route from the specified source to the specified destination.

Global Commands

All pass-through commands are preceded by an `@` symbol and a space.

The following client commands are supported:

`@ !`   disable status reporting
            Status reporting is disabled by default.

`@ ?`   enable status reporting
            Status reporting is enabled.

`@ Z:`   reset routing table

Routing is reset so that the first source is routed to all destinations.

Immediate Commands

`@ X:0/destination,source`   change route

`@ X:0/destination,source/destination-2,source-2`...   change multiple routes
            The specified source ports are routed to the specified destinations.
            Any routing changes will trigger `S:` notifications

`@ X?0destination`   request individual route status
            The source routed to a the specified destination will be returned as an `S:` notification.

`@ S?0`   request all ports route status
            Each source & destination port pair will be returned as `S:` notifications

Salvo Commands

`@ P:0/destination,source`   queue route change

`@ P:0/destination,source/destination-2,source-2`...   queue multiple route changes
            The specified routing changes are added to the current salvo for later execution.

`@ P?0destination`   request individual port status in salvo
            If a routing change for the specified destination port is queued, the route will be returned as a `V:` notification.

`@ V?0`   request all ports status in salvo
            Each queued routing change in the salvo is reported as a `V:` notification.

`@ B:C`   clear salvo

`@ B:R`   clear salvo
            Any queued changes are discarded and the salvo is reset.

`@ B:E`   execute salvo
            Any queued changes are executed and each routing change will be returned as an `S:` notification.
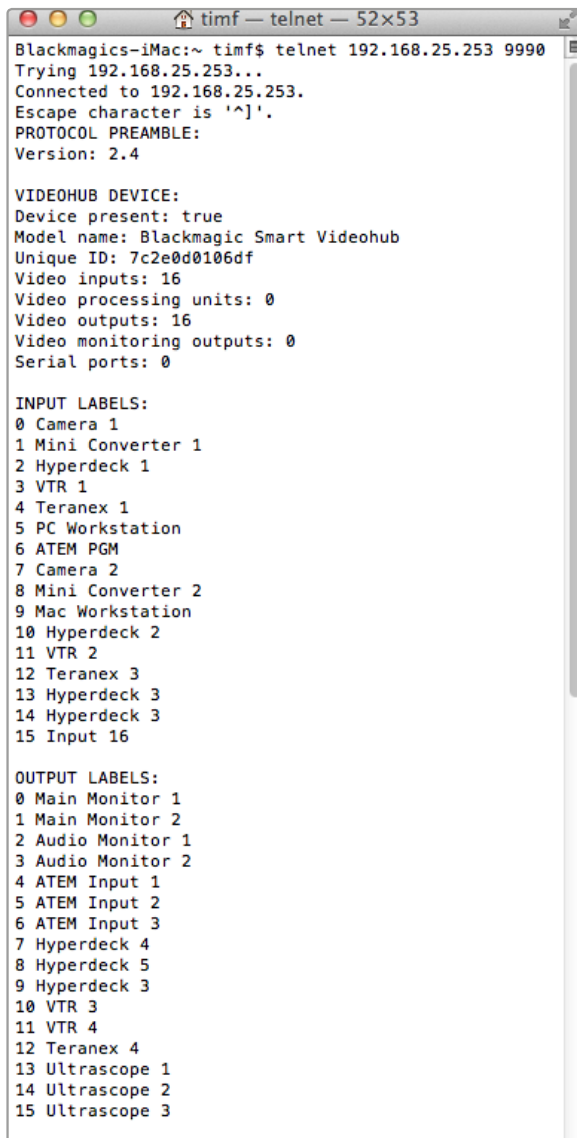
Saving and Loading Labels with Telnet in Mac OS X

Normally you would use Blackmagic Videohub Setup to save and load labels between different Videohub routers and to backup your videohub settings. However, if for any reason you wish to use the Terminal, the instructions below let you save and load router label configurations via the Terminal.

Saving Labels

1 Open the Terminal application which is located within the Applications > Utilities folder.

2 Type in "telnet" and a space followed by the IP of your Videohub, then another space and "9990", which is the default port number. For example type: telnet 192.168.25.253 9990.
The Protocol Preamble screen will appear.

3 Copy the INPUT LABELS: text including the heading and paste into a word processor or simple text editor. Name this file "input labels" and save to your favorite location.

4 Copy the OUTPUT LABELS: text including the heading and paste into a word processor or simple text editor. Name this file "output labels" and save to your favorite location.

5 Close the telnet session by exiting the terminal window.

Loading Labels

1 Open the Terminal application with is located within the Applications > Utilities folder.

2 Type in "telnet" and a space followed by the IP of your Videohub, then another space and "9990", which is the default port number. For example type: telnet 192.168.25.253 9990.
The Protocol Preamble screen will appear.

3 Open the "input labels" text file in your text editing program and copy the INPUT LABELS: text.

4 Paste this into the telnet session and press "return" twice. Telnet will respond with "ACK" and Videohub will update the input labels.

5 Open the "output labels" text file in your text editing program and copy the OUTPUT LABELS: text.

6 Paste this into the telnet session and press "return" twice. Telnet will respond with "ACK" and Videohub will update the output labels.

7 Close the telnet session by exiting the terminal window.

```
⊗⊗⊗            ⌂ timf — telnet — 52×53
Blackmagics-iMac:~ timf$ telnet 192.168.25.253 9990
Trying 192.168.25.253...
Connected to 192.168.25.253.
Escape character is '^]'.
PROTOCOL PREAMBLE:
Version: 2.4

VIDEOHUB DEVICE:
Device present: true
Model name: Blackmagic Smart Videohub
Unique ID: 7c2e0d0106df
Video inputs: 16
Video processing units: 0
Video outputs: 16
Video monitoring outputs: 0
Serial ports: 0

INPUT LABELS:
0 Camera 1
1 Mini Converter 1
2 Hyperdeck 1
3 VTR 1
4 Teranex 1
5 PC Workstation
6 ATEM PGM
7 Camera 2
8 Mini Converter 2
9 Mac Workstation
10 Hyperdeck 2
11 VTR 2
12 Teranex 3
13 Hyperdeck 3
14 Hyperdeck 3
15 Input 16

OUTPUT LABELS:
0 Main Monitor 1
1 Main Monitor 2
2 Audio Monitor 1
3 Audio Monitor 2
4 ATEM Input 1
5 ATEM Input 2
6 ATEM Input 3
7 Hyperdeck 4
8 Hyperdeck 5
9 Hyperdeck 3
10 VTR 3
11 VTR 4
12 Teranex 4
13 Ultrascope 1
14 Ultrascope 2
15 Ultrascope 3
```

Saving and Loading Labels with Telnet in Windows

While you can save and load labels within Blackmagic Videohub Setup, you may also download and use PuTTY which is a free telnet client.
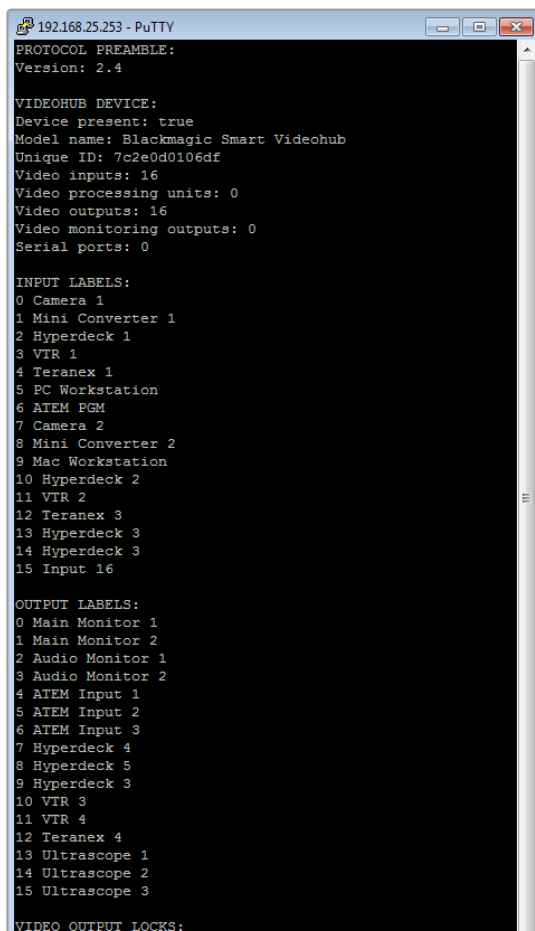
Setting up PuTTY

1   Start the application and in the "Host Name" field, type the IP of your Videohub and in the "Port" field type "9990", which is the default port number.

2   From the Connection type options select the "Telnet" radio button.

3   Select the "Terminal" tab and tick the "Implicit CR in every LF" checkbox.

4   Click the "Session" tab and save your session parameters by entering a name in the "Saved Sessions" field and clicking the "Save" button.

Saving Labels

1 Double click the PuTTY icon to start the application. Select your saved session from the list and click the "Load" button followed by the "Open" button. The Protocol Preamble screen will appear.

2 Copy the INPUT LABELS: text including the heading and paste into a word processor or simple text editor. Name this file "input labels" and save to your favorite location.

3 Copy the OUTPUT LABELS: text including the heading and paste into a word processor or simple text editor. Name this file "output labels" and save to your favorite location.

4 Exit the telnet session by closing the PuTTY window.

Loading Labels

1 Double click the PuTTY icon to start the application. Select your saved session from the list and click the "Load" button followed by the "Open" button. The Protocol Preamble screen will appear.

2 Open the "input labels" text file in your text editing program and copy the INPUT LABELS: text.

3 Right click in the PuTTY session to paste the copied section and press "Enter" twice. PuTTY will respond with "ACK" and Videohub will update the input labels.

4 Open the "output labels" text file in your text editing program and copy the OUTPUT LABELS: text.

5 Right click in the PuTTY session to paste the copied section and press "Enter" twice. PuTTY will respond with "ACK" and Videohub will update the output labels.

6 Exit the telnet session by closing the PuTTY window.