



PFD 4x4/4x2 Serial Protocol

Rev.1.06 07/22/2008

244 Bergen Blvd West Paterson NJ 07424 973-785-4347 www.fsrinc.com

Table of Contents

GENERAL NOTES	5
1.1 PFD 4x4/4x2 Model Types	5
1.2 Input Ranges:	5
1.3 Audio Ranges:	5
1.4 Special Characters:.....	5
1.5 Field Separators:.....	5
1.6 Command Request Syntax:	5
2 PFD 4X4/4X2 COMMAND LIST QUICK REFERENCE.....	6
3 REQUEST/RESPONSE FORMAT	6
4 ACKNOWLEDGING RECEIPT OF COMMANDS	6
5 ERROR RESPONSE	7
6 CONNECT REQUEST	8
7 DISCONNECT REQUEST	9
8 HELP REQUEST	10
9 MODEL REQUEST.....	13
10 MUTE REQUEST.....	14
11 PASSWORD REQUEST.....	15
12 RETURN-TO-FACTORY-DEFAULTS REQUEST	16
13 RESPONSE REQUEST	17
14 SET BAUD RATE REQUEST	18
15 STATUS REQUEST.....	19

16	VERSION NUMBER REQUEST	21
17	VOLUME REQUEST.....	22
18	VERBOSE REQUEST	23
19	PASS-THRU-PROTOCOL REQUEST.....	24

General Notes

1.1 PFD 4x4/4x2 Model Types

The PFD 4x4/4x2 is available with Video and Audio modules or Video module only. The user should take this into consideration as they review the syntax of commands and responses appearing below.

1.2 Input Ranges:

Inputs range from 1 to 4.

1.3 Audio Ranges:

Input trim value range -5dB to +15dB, output level range -45dB to +5dB.

1.4 Special Characters:

Carriage returns (Enter Key, 0Dh) will be referred to in the document as `<cr>`.
Linefeeds (0Ah) will be referred to in the document as `<lf>`.

1.5 Field Separators:

Fields are separated by *white* space, that is any number of spaces or tabs as long as the entire command is less than 80 characters. A `<cr>` terminates the command. Below is an example describing a (fictitious) command.

EX	05	<cr>
----	----	------

So the actual message would look like this:

EX 05<cr>

1.6 Command Request Syntax:

This document uses the following notation when describing the syntax of a command request:

- BOLD** – identifies the command
- lower case – identifies data to be entered which is described in the text following the syntax description
- “ ” - entry defined within double quotes is to be entered exactly as shown
- < > - entry defined within these brackets is required
- [] - entry defined within these brackets is optional
- | - a vertical bar denotes a logical choice of entry
- * - an asterisk following either type of bracket above denotes that data within brackets may be entered multiple times

2 PFD 4x4/4x2 COMMAND LIST QUICK REFERENCE

COMMAND	DESCRIPTION	PAGE
CON	Connect Request	8
DIS	Disconnect Request	9
HLP	Help – Request list with format	10
MOD	Model Request	13
MUT	Mute Request	14
PSW	Password Request	15
RFD	Return-to-Factory defaults	16
RSP	Response Request	17
SBR	Set Baud Rate	18
STA	Status Request	19
VER	Version Number Request	21
VOL	Volume Request	22
VRB	Verbose Error messages	23
{ }	Pass-thru-Protocol Request	24

3 Request/Response Format

All requests and responses will be entirely in ASCII. This will make the PFD 4x4/4x2 easy to use with the RN-8200, as well as other control systems from AMX, Crestron, etc.

All requests/responses will have a two to three character type field followed by the data required for that specific request/response. All requests may be terminated with a carriage return (0Dh), the typical terminator, which will be referred to in this document as *<cr>* or may also be terminated with a semi-colon character, “;”. The semi-colon character may also be used to separate successive commands, thereby allowing multiple commands on the same line. Responses are terminated with a carriage return and line feed (0Ah) *<lf>*.

4 Acknowledging Receipt of Commands

Each request sent to the PFD 4x4/4x2, except for the Pass-thru-Protocol request, will have two possible responses, one for the acknowledgement of a correct request and an error response. The normal acknowledge response will be “Ok” followed by a *<cr><lf>*. The *<cr><lf>* termination characters are understood in the responses shown below and are omitted.

Example:

Ok

5 Error Response

It is inevitable that errors occur in the requests sent to the PFD 4x4/4x2. If an invalid *command* is sent to the PFD 4x4/4x2, the PFD 4x4/4x2 will respond with the message “**ERR: unknown command**“. If an invalid parameter is sent to the PFD 4x4/4x2, the PFD 4x4/4x2 will respond with the message “**ERR:** “ followed by the valid syntax for the errored entry.

Example:

A connect request with an incorrect type and output number:

CON 1 R (8)

The error response would be:

ERR: CON {<input> <cardSpecifier> <\"(\"numberSequence\")\">}

Note that the returned error response may be expanded to a more detailed version by entering *verbose mode*. This mode may be turned on by issuing a **VRB Y** command. The expected syntax for the command will be returned, as above, and a description of each parameter.

For example, if verbose mode were turned on and the above command issued, the response would be:

ERR: CON {<input> <cardSpecifier> <\"(\"numberSequence\")\">}

Connect video (V), audio (A), or both (VA) of the specified input (1-4).

input	Input to connect "1" "2" "3" "4"
cardSpecifier	Card specifier "V", "A", or "VA" for video, audio or both.
outputSequence	Output as a number sequence.

6 Connect Request

The Connect request will instruct the PFD 4x4/4x2 to connect an audio input, video input or both to one or more outputs. The format for the Connect request is as follows:

CON	input	card specifier	“(“numberSequence”)”	<cr>
------------	-------	----------------	----------------------	------

Syntax:

CON {<input> <card specifier> <“(“numberSequence”)”>}* <cr>

Where:

CON Connection request header
input Input to connect “1” | “2” | “3” | “4”.
card specifier V for video, A for Audio, or VA for both.
numberSequence Outputs as a *number sequence*.

Example 1:

CON 3 A (1-4)<cr>

Instructs the PFD 4x4/4x2 to connect input 3 audio to outputs 1, 2, 3 and 4.

Example Response:

OK

This response means that the command has been received and is valid.

Example 2:

CON 4 VA (1,3)<cr>

Instructs the PFD 4x4/4x2 to connect input 4 video and audio to outputs 1 and 3.

Example Response:

OK

This response means that the command has been received and is valid.

Error Example:

CON 6 V (1)<cr>

Instructs the PFD 4x4/4x2 to connect input 6 video, which is invalid since there are only 4 inputs, to output 1.

Error Response:

ERR: CON {<input> <cardSpecifier> <“(“numberSequence”)”>}*

7 Disconnect Request

The Disconnect request is used to disconnect a specified video output, audio output, or both from its input.

DIS	output cardSpecifier[","output cardSpecifier]	<cr>
------------	-----------------------------------------------	------

Syntax: DIS <output> <cardSpecifier>[","<output> <cardSpecifier>]*<cr>

Where:

DIS Disconnect request header
output Output being disconnected.
cardSpecifier Optional card specifier "V", "A", or "VA" for video, audio, and video plus audio respectively.

Example:

To disconnect video outputs 1, 3, 4 the user would send the following message:

`DIS 1 V, 3 V, 4 V<cr>`

To disconnect video and audio outputs 1, 3, and 4 the user would send the following message:

`DIS 1 VA, 3 VA, 4 VA<cr>`

To disconnect video output 3 and audio output 4 the user would send the following message:

`DIS 3 VA, 4 A<cr>`

If the user attempted to disconnect a video output without the proper card specifier then an error would be reported. In the example below, the user attempts to disconnect output 1 without specifying the card

`DIS 1<cr>`

`ERR: DIS <output> <cardSpecifier>[","<output> <cardSpecifier>]*<cr>`

8 Help Request

The user will have the ability to get a list of all commands from the PFD 4x4/4x2. The user will issue the following request:

HLP	[cmd]	<cr>
-----	-------	------

Syntax: HLP[cmd]<cr>

Where:

HLP Help Request header
cmd optional command identifier

If the optional **cmd** is omitted, the PFD 4X4/4X2 will respond with the following text message:

Response (*cmd* omitted):

CON Connect audio, video, or both of a specified input to specified outputs.
DIS Disconnect request.
HLP Provides help information for PFD 4X4/4X2 command set.
MOD Model number identification.
MUT Mute or unmute audio for the specified outputs.
PSW Secure data access request.
RSP Request or decline a response.
STA Request status of video and audio connections and volumes.
VER Request current part # and version number.
VOL Adjust audio output level or input trim levels.
VRB Turns on or off verbose command error reporting.
{ } Forwards a string to specified logical link (see manual).

Entering **HLP <cmd><cr>**, where *cmd* is any valid PFD 4X4/4X2 command in the above list, will return specific help for the command requested.

Specific command help is listed below:

Response (*cmd* included):

CON {<input> <card specifier> <"(numberSequence)">}*<cr>
Connect video (V), audio (A), or both (VA) of the specified input (1-4).
input Input to connect "1"|"2"|"3"|"4"
cardSpecifier Card specifier "V", "A", or "VA" for video, audio or both.
outputSequence Output as a number sequence.

DIS <output> <cardSpecifier>[", "<output> <cardSpecifier>]*<cr>
Disconnect specified outputs.
output Output being disconnected.
cardSpecifier Card specifier "V", "A", or "VA" for video, audio or both.

HLP[cmd]<cr>
Provides help information for the PFD 4X4/4X2 command set.

cmd optional command identifier

MOD <""modelType""><cr>

Specifies the model type of the matrix.

""modelType"" The model type string per pre-specified allowed values.

MUT <"M"|"U"> <outputsequence><cr>

Mutes or unmutes audio for the specified outputs.

"M"|"U" "M" = mute, "U" = unmute

outputsequence Output as a number sequence

PSW <""password""><cr>

Login using previously specified password.

password The preconfigured FSR password which allows access to reserved features. The password must be enclosed within single or double quotes.

RSP <"Y"|"N"><cr>

Enables or disables a response from PFD 4X4/4X2.

"Y"|"N" Y = response enabled, N = response disabled

STA<cr>

Status request returns the following:

If STA I specified:

STA I

1 <TrimInp1>

2 <TrimInp2>

3 <TrimInp3>

4 <TrimInp4>

If STA O specified:

STA O

<Number"V"> <Number"A"> <MuteStatus> <OutputLevel>

for each of outputs

Number Input (1-4)

MuteStatus "M" = Muted, "U" = Unmuted

OutputLevel output level -45 to +5 in dB

TrimInp audio trim -5 to +15 in dB for each of 4 inputs

VER<cr>

Version request returns the following:

VER Part# Revision#

Part# "PFD-4x4-PCA" | "PFD-4x4-PC" | "PFD-4x2-PCA" | "PFD-4x2-PC"

Revision# XX.xx, XX=Major version number, xx=Minor version number

VOL <<"L"|"T" number> <value|"?"><cr>

"L" | "T" output Level or Trim

number values of "1" to "4" applicable only when "T" designated
value volume value -5 to +15 for trims, -45 to +5 for level (in 1 dB
increments) or "+" to increment volume in 1 dB increments, or "-" to
decrement volume in 1 dB decrements.
"?" inquiry for the specified level or trim

VRB <"Y"|"N"><cr>

Turns on or off verbose error message reporting.

"Y"|"N" "Y" turns verbose on, "N" turns verbose off.

9 Model Request

The MOD Model request allows configuration of the model type of the matrix. This command is normally reserved for use by FSR technical personnel and as such the user must have logged in correctly with the PSW command. The format for the model request is as follows:

MOD	<code>""modelType"" ?"</code>	<code><cr></code>
------------	-------------------------------	-------------------------

Syntax: MOD `<""modelType""|?"><cr>`

Where:

MOD

Model request header.

`"?"`

Inquire the Pathfinder matrix model number. Return string is in format as described below.

`""modelType""`

The model type in a format defining the number of inputs and outputs, the video type, audio card presence, and/or equalization present. The modelType must be enclosed within single or double quotes. The modelType must be *exactly* one of the following:

PFD-4x4-PC
PFD-4x4-PCA
PFD-4x2-PC
PFD-4x2-PCA

Example:

To configure the matrix model type as 4x4 with audio present would issue the following string from the list above.

`MOD "PFD-4x4-PCA"<cr>`

To obtain the model number the user would issue the following command:

`MOD ?<cr>`

For a 4x2 without audio functionality the response will be:

`PFD-4x2-PC`

10 Mute Request

The user will be able to mute any audio output by sending the mute request:
The format for the mute request is as follows:

MUT	"M" "U"	outputSequence	<cr>
------------	-----------	----------------	------

Syntax: MUT <"M"|"U"> <outputSequence><cr>

Where:

MUT Start of mute message

"M"|"U" M=mute, U=unmute

outputSequence Output as *number sequence* to be muted/unmuted.

Example:

MUT M 4<cr> – Mutes the audio output of output 4.

MUT U 3<cr> – Unmutes the audio output of output 3.

MUT M 1-4<cr> Mutes audio outputs 1, 2, 3, and 4.

11 Password Request

The PSW Password request allows access to additional special commands including, but not limited to, the MOD command to configure the model type of the matrix unit. This command is generally reserved for use by FSR technical personnel only. Enhanced access returns to the limited command set on a power cycle.

PSW	<code>""password""</code>	<code><cr></code>
------------	---------------------------	-------------------------

Syntax: PSW `<""password""><cr>`

Where:

PSW PSW password request header
password The preconfigured FSR password which allows access to reserved features. The password must be enclosed within single or double quotes.

Example:

To prepare the matrix for access to a special command set the user would send the following message:

`PSW "password"<cr>`

The matrix will respond with an Ok message.

12Return-to-Factory-Defaults Request

The user may return the unit to default factory settings by issuing this command. (This command is generally intended for use by FSR personnel only and is password protected.) The format for the Return-to-Factory-Defaults request is as follows:

RFD	"Y" "N"	<cr>
------------	---------	------

Syntax: RFD <"Y"|"N"><cr>

Where:

RFD Return-to-Factory Defaults Request header.

"Y"|"N" Y = Return to factory defaults, N = Do not return to factory defaults.

Example:

RFD Y<cr> – Return to factory defaults requested.

RFD N<cr> – Remain as configured.

13 Response Request

The user will be able to suppress all responses from the PFD 4X4/4X2 if they like. The user should ensure that all subsequent requests to the PFD 4X4/4X2 are properly formatted as it would be difficult to ascertain their correctness without a response. The format for the Response request is as follows:

RSP	"Y" "N"	<cr>
------------	---------	------

Syntax: RSP <"Y"|"N"><cr>

Where:

RSP Response Request header.

"Y"|"N" Y = response enabled, N = response disabled

Example:

RSP Y<cr> – Response requested.

RSP N<cr> – Response suppressed.

14 Set Baud Rate Request

The user may alter the serial bit rate of the downstream link, SCI 2, by issuing this command. The downstream link is used to forward serial commands received on SCI 1 to equipment connected to SCI 2. Speeds accommodated, in bits per sec, are as described below. *The default configuration is 38,400 bps.* The format for the Set Baud Rate request is as follows:

SBR	"1" "2" "3" "4" "5" "6" "?"	<cr>
------------	-----------------------------	------

Syntax: SBR "1"|"2"|"3"|"4"|"5"|"6"> | "<?"><cr>

Where:

SBR Set Baud Rate Request header.

"1"|"2"|"3"|"4"|"5"|"6" 1=2400, 2=4800, 3=9600, 4=19200, 5=38400, 6=57600

"?" Inquiry returns the currently configured baud rate.

Example:

SBR 1<cr> – Sets serial baud rate to 2400 bps.

SBR ?<cr> – If 2400 bps was previously configured, this command returns:

1=2400 bps

15 Status Request

The user may request status from the PFD 4X4/4X2 at any time. For a request for input status, the returned status will be the current setting for each input in dB. For a request for output status, the returned status will be each output's video input connection, audio input connection, mute status and output level in dB. The format of the Status request is as follows:

STA	"I" "O"	<Cr>
------------	-----------	------

Syntax: STA<cr>

Where:

STA Status Request header

Response for Output Status request:

STA O

<Number"V"> <Number"A"> <MuteStatus> <OutputLevel>

Where:

Number Values **1, 2, 3, or 4** representing the current Video input or Audio input.

MuteStatus **M** for muted, **U** for unmuted

OutputLevel Value representing the current Audio Output Level in range –45 to +5.

The above status is repeated for each of the 4 (PFD 4x4) or 2 (PFD 4x2) outputs.

Response for Input Status request:

STA I

<TrimInp1>

<TrimInp2>

<TrimInp3>

<TrimInp4>

Where:

TrimInp Value representing the current Audio Trim Level in range –5 to +15 for each of 4 Audio sources.

Example:

STA O<cr>

Requests output status of the PFD 4X4/4X2.

Example Response:

STA O

1V 1A U -20

2V 2A U -30

3V 3A M 0

4V 4A U +5

This response shows that input 1 is connected to output 1 video and audio output, is unmuted and set to –20 dB; input 2 is connected to output 2 video and audio output, is unmuted and set to –30 dB; input 3 is connected to output 3 video and audio output, is

muted and set to 0 dB; input 4 is connected to output 4 video and audio output, is unmuted and set to +5 dB.

Example:

```
STA I<cr>
```

Requests input status of the PFD 4X4/4X2.

Example Response:

```
STA I
```

```
1 +15
```

```
2 -5
```

```
3 +0
```

```
4 +3
```

This response shows that inputs 1, 2, 3, and 4 are set to +15 dB, -5dB, 0 dB, and +3 dB respectively.

16Version Number Request

The user has the ability to request the PFD 4X4/4X2's Part Number and the current firmware version number. The request is formatted as follows:

VER	<cr>
------------	-------------------

Syntax: VER<cr>

Where:

VER Version Request header

Response: **VER** *Part#* *Version#*

Where:

Part# **PFD-4x4-PCA** or **PFD-4x2-PCA** – for Video with Audio option

PFD-4x4-PC or **PFD-4x2-PC** – for Video without Audio option

Version# XX.xx, where XX = Major version number and xx = Minor version number.

Example:

VER<cr>

To which the PFD 4X4 Video and an Audio Option and firmware revision 1.04 will respond with the following:

VER PFD-4x4-PCA 01.04

17 Volume Request

The Volume request is used to adjust input trims or output level. It can be used to inquire about the current value of an input trim or output level. The trims and levels can be set as a value, or adjusted up or down with “+” or “-”.

VOL	“L” “T” number	value ”?”	<cr>
------------	------------------	-----------	------

Syntax: VOL <<”L”|”T”> <number> <value|”?”>><cr>

Where:

VOL Volume request Header

“L” | “T” output Level or Trim

number The output number if L specified or the input number if T specified.

value volume value -5 to +15 for trims, -45 to +5 for level (in 1 dB increments) or “+” to increment volume in 1 dB increments, or “-” to decrement volume in 1 dB decrements. Multiple “+” or “-” may be specified consecutively, each representing 1 dB step.

“?” inquiry for the specified level or trim

Example:

```
VOL L 1 23<cr>
```

Instructs the PFD 4X4/4X2 to set the output level out output 1 to 23 dB.

Example Response:

```
OK
```

This response means that the command has been received and is valid.

Error Example:

```
VOL T 2 40<cr>
```

Instructs the PFD 4X4/4X2 to set input 2’s trim to “40” which is invalid.

Error Response:

```
ERR: <<”L”|”T”> <number> <value|”?”>>
```

The PFD 4X4/4X2 would send the error message above indicating that the “40” in the message was invalid (trim values are from -5dB to +15dB).

VOLUME INQUIRY

If the user wishes to know what trim or level is set at, the user can insert a “?” in place of the Value, an Inquiry will look like this:

Example:

```
VOL L 1 ?<cr>
```

This requests the output level of output 1, to which the PFD 4X4/4X2 will respond:

Example Response:

```
VOL L1 -10
```

This response informs the user that the current output level setting of output 1 is -10dB.

18 Verbose Request

The user can, during debugging, turn on verbose mode. This mode will expand the standard error messages to include the expected command syntax along with a description of the commands and possible parameters as formatted above in the specific help for that errored command. If an input command is unidentified, verbose mode will output the contents of the general HLP command. If verbose mode is off, then any errored input will result in just the command syntax being output as described in **Syntax** for each command. The verbose request is defined as follows:

VRB	"Y" "N"	<cr>
------------	---------	------

Syntax: VRB <"Y"|"N"><cr>

Where:

VRB Verbose request header
"Y"|"N" "Y" turns verbose on, "N" turns verbose off.

Example:

If the user tries to connect input 6 (which is not in valid range of 1-4) using the CON command, the error message would appear as follows:

ERR: CON {<input> <cardSpecifier> <\"numberSequence\">}

Connect video (V), audio (A), or both (VA) of the specified input (1-4).

input Input to connect "1"|"2"|"3"|"4"
cardSpecifier Card specifier "V", "A", or "VA" for video, audio or both.
outputSequence Output as a number sequence.

19 Pass-thru-Protocol Request

The Pass-thru-Protocol facilitates reaching end devices not able to be directly connected with the originator of a command.

The PFD 4x4/4x2 is capable of forwarding a serially received request to a device further downstream using the Pass-thru-Protocol. The request to be forwarded is essentially composed of two parts. The first part, an ASCII string referred to as the Drop Preamble, describes the logical link to which the second part, it too an ASCII string, will be forwarded to. For the PFD 4x4/4x2, the logical link may only be a serial link but devices further downstream may have the capability to designate other types of logical links that can be incorporated in the contents of the drop preamble. The Pass-thru-Protocol request is defined as follows:

“{“Drop Preamble”}”	“””Drop String”””	<cr>
---------------------	-------------------	------

Note: the syntax description below follows our convention of characters within quotes as being required, contents enclosed within [] brackets as being optional and the * character meaning that it may be repeated 1 or more times. Also note that the syntax is presented as a BNF grammar to fully specify the syntax. This form of grammar decomposes the initial description of the Drop String into progressively smaller units that identify the full extent of the composition of characters allowed. Examples are provided below for clarification.

Syntax: “{“Drop Preamble”}” “””Drop String”””<cr>

Where:

- Drop Preamble [“\S[#]”]*, That is: a \ character, followed by a S character identifying a *serial* logical link, followed by an optional number designating which serial logical link. That sequence may be repeated as many as 5 times, a practical limit.
 For example, \S\S1\S2\S\S2.
 This designates a drop preamble that will traverse five logical links (each which must have Pass-thru-Protocol capability) before reaching the end device to which the Drop String is intended to target (further examples later).
- Drop String Drop String is enclosed within double quote characters.
 = [SpecialASCIISeq] [Non-Special ASCII Char] [ASCIISHexSeq]
 DropString
- SpecialASCIISeq = “\<” | “\>” | “\” | “\”” (ie a backslash char, \, precedes the special characters <, >, \, and “, which have special uses within the Drop String (see below).
- Non-Special ASCII Char = ASCII characters
- ASCIISHexSeq = “<” [ASCIISHexBytePair]*>”, ie zero or more pairs of ASCII hex bytes separated by a space and enclosed within <> characters.
 For example, <AA BB CC 0F 0D>.
- ASCIISHexBytePair = XX, where X = 0-9, A-F.

Example:

The target device to which the Drop String contents are intended may be a EIKI projector and the user may wish to turn it on, which is the sequence C00 followed by a carriage return. If the projector is directly connected to the downstream serial port of the PFD 4x4/4x2 then the Pass-thru-Protocol request will be the following:

```
{\S} "C00 <0D>"<cr>
```

The above request has the drop preamble directing the Drop String enclosed within double quotes to the PFD 4x4/4x2 downstream serial link. The PFD 4x4/4x2 will interpret the drop preamble as this device being the final forwarding device and will send the string C00 <cr> (the 0D ASCII hex pair represents a carriage return) to its downstream serial link.

If the projector were connected via two successive PFD 4x4/4x2 devices, or perhaps a single PFD 4x4/4x2 and a second Pass-thru-Protocol capable device then the request would be the following:

```
{\S\S} "C00 <0D>"<cr>
```

If the end target device, that is the one receiving the Drop String, is attached to a PtP capable device via 4 intervening PtP capable devices and contains a combination of both ASCII characters and ASCII hex sequences then the (fictitious) command might look as follows:

```
{\S\S\S\S\S} "A more complex example <AA BB 55 AB> with multiple embedded ASCII hex seqs <01 02 FF 0D>"
```

The above request addresses 5 PtP capable devices. The first device interprets the Drop Preamble directing the Drop String to its serial logical link, S. This device strips off its logical link, the first \S, and sends the remainder out serial logical link S, ie {\S\S\S\S\S} "A more complex example <AA BB 55 AB> with multiple embedded ASCII hex seqs <01 02 FF 0D>".

Each succeeding device interprets the Drop Preamble destination, strips off its address and forwards the remaining request. This occurs until the last device in the Drop Preamble in which the remaining request is: {\S} "A more complex example <AA BB 55 AB> with multiple embedded ASCII hex seqs <01 02 FF 0D>".

The last device will strip off the Drop Preamble entirely (as nothing remains of it after stripping off \S and parse the Drop String. The parsing will output any non-special sequences as is and will interpret any ASCII hex sequences and output the corresponding hex values. Thus for our example above, the final output is:

A more complex example 0xAA 0xBB 0x55 0xAB with multiple embedded ASCII hex seqs 0x01 0x02 0xFF 0x0D

Where the hex sequences 0xAA, 0xBB, etc are raw hex values and NOT ASCII hex.

The response to any PtP request by the end device will be returned upstream via any and all intervening PtP capable devices to the originating device. Thus, if there were a response by the EIKI projector to the Power On command, it will be forwarded to the originator.