



Epiphan Video

Pearl-2, Pearl Mini, Pearl Nano, Pearl Nexus

REST API Crestron Driver
ClearAlan Inc.

Last Revision: August 22, 2026



www.ClearAlan.ca



877-850-2871

Revision History

Version	Date	Comment	Author
v1.0.0	2026-08-22	Initial Release	SP & TA

Table of Contents

Dependencies	7
General Notes	7
Setup of Crestron Hardware	7
Quick Start Guide	7
Signal Map	9
1 Parameters:	9
2 Setup	9
Inputs:	9
3 Connection:	10
Input:	10
Feedback:	11
4 Recorder:	12
Input:	12
Feedback:	12
Recorder[*] Inputs – 1 to 8:	13
Recorder[*] Feedbacks – 1 to 8:	13
5 Channel:	15
Feedback:	15
Channel[*] Inputs - 1 to 8:	15
Channel[*] Feedbacks - 1 to 8:	16
Channel[*] Inputs - Publisher[n] (n = 1 to 5):	18
Channel[*] Feedbacks - Publisher:	18
Channel[*] Feedbacks - Publisher[n] (n = 1 to 5):	18
Channel[*] Input - Layout	20
Channel[*] Inputs - Layout[n] (n = 1 to 10):	20
Channel[*] Feedbacks - Layout:	20
Channel[*] Feedbacks - Layout[n] (n = 1 to 10):	20
Channel[*] Feedback - AudioLevel:	21
6 Inputs	22
Input[*] Inputs – 1 to 25:	22

	Feedback:	23
	Input[*] Feedback – 1 to 25:	23
7	Outputs	25
	Output[*] Inputs - 1 to 5:	25
	Feedback:	25
	Output[*] Feedbacks – 1 to 5:	25
8	CMS Kaltura	29
	Inputs:	29
	Feedback:	30
9	CMS Panopto	32
	Inputs:	32
	Feedback:	33
10	CMS Yuja	35
	Inputs:	35
	Feedback:	35
11	CMS Opencast	37
	Inputs:	37
	Feedback:	38
12	CMS Epiphan Edge	40
	Inputs:	40
	Feedback:	40
13	CMS Echo360	42
	Inputs:	42
	Feedback:	43
14	CMS Generic Container	45
	Inputs:	45
	Feedback:	46
15	Events	48
	Feedback:	48
	Event.Upcoming - Inputs:	48
	Event.Upcoming - Feedback:	48
	Event.Current - Inputs:	49
	Event.Current - Feedback:	50

	Event.Create - Inputs:	51
	Event.Create - Feedback:	51
	ScheduledEvent[*] Inputs – 1 to 10:	52
	ScheduledEvent Feedback:	52
	ScheduledEvent[*] Feedback – 1 to 10:	52
	PastEvent Feedback:	53
	PastEvent[*] Feedback – 1 to 10:	53
16	Presets	55
	Inputs:	55
	Feedback:	55
17	System	57
	Inputs:	57
	Feedback:	57
	System.Disk.Internal Feedback:	58
	System.Disk.External Inputs:	59
	System.Disk.External Feedback:	59
18	Device	61
	Feedback:	61
19	Network	62
	Feedback:	62
20	Debug	63
	Inputs:	63
	Feedback:	63
21	Stream Management	64
	Inputs – Stream.Management:	64
	Inputs – Stream.Management.RTSP:	64
	Inputs – Stream.Management.RTMP:	65
	Inputs – Stream.Management.RTP_UDP:	65
	Inputs – Stream.Management.MPEG-TS_UDP:	65
	Inputs – Stream.Management.MPEG-TS_RTP:	66
	Feedback:	66
22	Error	67
	Feedback:	67

Complementary Modules	68
1 Construct Test Field Debouncer	68
Inputs:.....	68
Parameters:	68
Feedback:	68
2 Epiphan Pearl Event List Sorting.....	69
Inputs:.....	69
Feedback:	69
3 Epiphan Pearl Input List Sorting	70
Inputs:.....	70
Feedback:	71
4 Epiphan Pearl Touchpanel Preview URL helper	72
Inputs:.....	73
Feedback:	74
Technical Tips and Tricks	75
1 Using the module – different device models, same module	75
Pearl-2	75
Pearl Mini.....	75
Pearl Nexus	76
Pearl Nano	76
2 Preview images	76
Troubleshooting.....	79
1 A secure connection to the device is required	79
2 Cannot connect to the device	79

Dependencies

This module requires v200 or later of the Crestron Database and Crestron Device Database.

The construct file used for the demo program was built on CH5 Version 2.16.1.

The module was tested on a CP3 and CP4 against an Epiphan Pearl-2 and Pearl-Nano running firmware version 4.24.4. The Epiphan Pearl-Mini and Pearl Nexus are also supported.

Although previous Pearl firmware versions may work, they will not be supported.

General Notes

This module is used for connecting an Epiphan Pearl device to control the peripheral through a control system.

There are two demonstration programs available to showcase the module: *Demo_EpiphanPearl* and *Demo_EpiphanPearl_3series*, alongside a UI program *Demo_EpiphanPearl.ch5z*. The two SIMPL Windows programs are identical except for their target rack. The Crestron module queries the type of Epiphan Pearl it is connected to and responds appropriately; thus, the same module and demonstration program can be used for any model of Epiphan Pearl.

Load the applicable .lpz file to your control processor of choice and load the project either onto the processor as an HTML5 Web XPanel or on any 10" x60, x70, or x80 series Crestron touchpanel, connected at IPID 03.

If running as an HTML5 Web XPanel, ensure that your processor is up to date, and is compatible with CH5 web XPanels.

Setup of Crestron Hardware

The module is supported on both 3- and 4-series Crestron processors. When using the module as part of a program running on a 3-series processor, a 3-series firmware version of at least 1.8001 or newer is recommended, and the latest version available should be used for the best results.

This SIMPL module allows for preview images to be visible on the user interface running the Crestron Construct demo UI project. When utilizing Crestron Construct in conjunction with preview images, a firmware version of at least 1.8001 or new is *required* for the preview images to function. See the Technical Notes section at the end of this document for more information about this.

Quick Start Guide

- 1) Ensure the **IP Address or HostName** parameter contains the IP address of the Epiphan Pearl.
- 2) Ensure the **Username** and **Password** fields have been set correctly. The default username is "admin" and the password is set on first login to the Pearl web UI.

- 3) In your program, latch the “Enable” signal when you would like the module to initiate the connection with the device. The module manages the connection to the device on its own; it will attempt to reconnect to the API if the connection is broken and regularly polls the device to ensure external changes in device state are reflected on the output signals of the module accordingly.
- 4) As previously mentioned, the module is the same for each model of Epiphan Pearl. Not all signals are applicable to all models. Consult the “Technical Tips” section outlined towards the end of this document for an in-depth explanation of which signals do not apply to all Epiphan Pearl modules.
- 5) Should difficulties arise with connecting to the device, consult the troubleshooting steps outlined at the end of this help file.

Signal Map

1 Parameters:

Name	Type	Description	Notes
IP Address or HostName	String	IP address or hostname for connection.	If specifying this value at runtime is needed, the <i>IP_Address</i> signal can be used to replace this value.
Port	Decimal	Default Port for connection	If specifying this value at runtime is needed, the <i>Port</i> signal can be used to replace this value.
Username	String	Default username for connection	If specifying this value at runtime is needed, the <i>Username</i> signal can be used to replace this value.
Password	String	Default password for connection	If specifying this value at runtime is needed, the <i>Password</i> signal can be used to replace this value.
UseHttps	Yes/No	Choose whether HTTPS or HTTP is used to connect with the Pearl.	If specifying this value at runtime is needed, the <i>UseHttps</i> or <i>UseHttp</i> signal can be used to replace this value.

2 Setup

Inputs:

Name	Type	Description	Notes
Enable	Digital	Latch this signal high to enable the IP connection. The module will disconnect and stop all activity on a trailing edge.	If using any of the input signals in the Connection signal group to specify the connection details at runtime, it is recommended that this signal be latched high <i>after</i> those values are set.

3 Connection:

Input:

Name	Type	Description
Connection.UseHttps	Digital	Configures the module to use HTTPS to communicate with the device on a rising edge. Will supersede the setting on the parameter field. HTTPS communication must be enabled on the Pearl to use HTTPS.
Connection.UseHttp	Digital	Configures the module to use HTTPS to communicate with the device on a rising edge. Will supersede the setting on the parameter field. HTTPS communication must be not enabled (this is the default) on the Pearl to use HTTP.
Connection.UseHttpOrHttps.Toggle	Digital	Toggles between using using HTTPS or HTTP to communicate with the device on a rising edge.
Connection.Username	String	The username used by the module to connect to the device. Will supersede the value on the parameter field.
Connection.Password	String	The password field used by the module to connect to the device. Will supersede the value on the parameter field.
Connection.IP_Address	String	The IP address (or hostname) of the device. Will supersede the value on the parameter field. If using hostnames, ensure the DNS setup on the AV network will resolve the hostname/FQDN properly.
Connection.Port	Analog	The port of the driver used for communication represented as an analog signal. Will supersede the value on the parameter field.
Connection.Port_String	String	The port of the driver used for communication represented a serial signal. Will supersede the value on the parameter field.

Feedback:

Name	Type	Description
Connection.UseHttps.Feedback	Digital	A high value indicates that the control system will attempt to communicate via HTTPS.
Connection.UseHttp.Feedback	Digital	A high value indicates that the control system will attempt to communicate via HTTP.
Connection.Connected	Digital	A high value indicates the control system has connected to the camera and is able to communicate properly with the REST API on the device.
Connection.Connection_Status	String	A string value from the module to display an accurate connection status from the control system to the peripheral.
Connection.Username.Feedback	String	The username used by the module to connect with the device.
Connection.Password.Feedback	String	The (unmasked) password used by the module to connect with the device.
Connection.IP_Address.Feedback	String	The IP address or hostname used by the module to connect with the device.
Connection.Port.Feedback	String	The TCP port number used by the module to communicate with the REST API, represented as an analog signal.
Connection.Port_String.Feedback	Analog	The TCP port number used by the module to communicate with the REST API, represented as a serial signal.

4 Recorder:

At the time of writing, the Epiphan Pearl has a recorder assigned to each channel configured on the device. Recorders 1- N are the recorders used for channels 1- N on the device. Recorder numbers $N+1$ and onwards are multitrack recorders, if the device has been configured to use them. This can cause some signal duplication with the recorder controls available in the Channel: section below, but allows for the greatest flexibility should the relationship between recorders and channels change in the future.

The module supports a maximum of 8 recorders between those assigned to a channel and multitrack recorders. Should your application have a need for more than this, please contact Epiphan with a feature request to have support for more added.

Input:

Name	Type	Description
Prefix	String	Specifies the prefix for all recording names. To remove prefix, simply send an empty string (ex: "")

Feedback:

Name	Type	Description
Recorder.Count.Feedback	Analog	Indicates the number of recorders configured on the unit. A recorder is assigned for each channel configured on the device, as well as for each multitrack recorder configured. This will only display the recorders on the device, not the maximum number of recorders available.

Recorder[*] Inputs – 1 to 8:

Name	Type	Description
Recording.Start	Digital	Start recording on the specified recorder on a rising edge.
Recording.Stop	Digital	Stop recording on the specified recorder on a rising edge.
Recording.Toggle	Digital	Toggle the recording state between started and stopped for the specified recorder on a rising edge.

Recorder[*] Feedbacks – 1 to 8:

Name	Type	Description
IsAvailable	Digital	A high value indicates a recorder at that index is configured on the device. Use this for conditional behavior depending on how many recorders are configured to display controls or take action accordingly.
Name	String	Indicates the user-configurable recorder name for the recorder. If the recorder is associated with a channel, this will be the user-configurable name for the channel instead.
State	String	Indicates the recorder state as a user-friendly description. Values: "Disabled", "Starting", "Started", "Stopped", "Error"
State.Feedback	Analog	Specifies the recorder state as an analog value, where: 0 = Disabled 1 = Starting 2 = Started 3 = Stopped 4 = Error In normal operation, the "Started" and "Stopped" values are observed when using the <i>Recording.Start</i> and <i>Recording.Stop</i> signals.
Duration	String	Specifies the current duration the recording has been active in a hh:mm:ss format.

IsMultiSource.Feedback	Digital	A high value indicates the recorder is a multitrack recorder.
IsRecording.Feedback	Digital	A high value specifies if the recorder has started recording (corresponds with the “Started” value for <i>State</i> above).
IsRecording.Feedback*	Digital	The complement of “ <i>IsRecording.Feedback</i> ” to show that the recorder is not recording

5 Channel:

This module supports a maximum of 8 channels, with a maximum of 5 publishers configured for each channel and 10 layouts for each channel. Note that by default, a channel has no publishers configured, and only 1 layout. Some Pearl models only support 1 channel and 1 layout on that channel; see the Technical Tips section towards the end of this document for details about this.

Should your application have a need for more than this, please contact Epiphan about having support for more added.

Feedback:

Name	Type	Description
Count.Feedback	Analog	Specifies the number of channels configured on the unit. This will only display the channels on the device, not the maximum number of channels available.

Channel[] Inputs - 1 to 8:*

Name	Type	Description
Preview.Enable	Digital	Latch high to enable retrieving preview images for the channel from the Pearl device. This specifically enables the preview image for the live channel, not any of the channel layouts themselves. Preview images are retrieved every 5 seconds from the device for each preview image item enabled. Generating preview images is a computationally expensive for the Epiphan Pearl device. Epiphan recommends minimizing the amount of preview images being generated where possible. It is recommended to review the CPU load metrics indicated by the Epiphan Pearl webpage during a “worst case” scenario while commissioning the system to determine if the device is still operating within its limits during this scenario. For more details about preview images, consult the “Technical Tips” section at the end of this document.
Recording.Start	Digital	Start recording for the specified channel on a rising edge.
Recording.Stop	Digital	Stop recording for the specified channel on a rising edge.
Recording.Toggle	Digital	Toggle the recording state between started and stopped for the specified channel on a rising edge.
AllStreams.Start	Digital	Starts the stream on all configured publishers for the channel on a rising edge.

AllStreams.Stop	Digital	Stops the stream on all configured publishers for the channel on a rising edge.
AllStreams.Toggle	Digital	Toggles all configured publishers' streaming state for the channel on a rising edge. In scenarios where not all publishers for the channel are started/stopped, the first configured publisher status is used to indicate the state to toggle to.

Channel[] Feedbacks - 1 to 8:*

Name	Type	Description
IsAvailable	Boolean	A high value indicates a channel at that index is configured on the device. Use this for conditional behavior depending on how many recorders are configured to display controls or take action accordingly.
Name	String	Indicates the user-configurable name for the channel
PreviewUrl	String	Indicates the preview URL for the channel. This is provided in a format that is not a complete URL, rather it describes the path of the resource for the preview image on the Crestron processor webserver that the module is running on. See the Technical Tips section towards the end of this document that describes how the processor must be configured and how to build the full URL to pass to the user interface.
Recording.Feedback	Digital	A high value specifies if the recorder has started recording (corresponds with the "Started" value for <i>Recorder.State</i> below).
Recording.Feedback*	Digital	The complement of " <i>Recording.Feedback</i> " to show that the channel is not recording
Recorder.State	String	Indicates the channel recorder state as a user-friendly description. Values: "Disabled", "Starting", "Started", "Stopped", "Error"
Recorder.State.Feedback	Analog	The analog value to represent the recorder state of the channel, where: 0 = None

		1 = Disabled 2 = Starting 3 = Started 4 = Stopped 5 = Error
Recorder.Duration	String	Specifies the current duration the channel recorder has been active in hh:mm:ss format.
AllStreams.Active.Feedback	Digital	A high value indicates all publishers are currently streaming for the channel
AllStreams.Active.Feedback*	Digital	The complement of "AllStreams.Active.Feedback" to show that all streams are not active

Channel[] Inputs - Publisher[n] (n = 1 to 5):*

Name	Type	Description
Start	Digital	Starts the stream on the specified publisher for the channel on a rising edge.
Stop	Digital	Stops the stream on the specified publisher for the channel on a rising edge.
Toggle	Digital	Toggles the stream between started and stopped on the specified publisher for the channel on a rising edge.

Channel[] Feedbacks - Publisher:*

Name	Type	Description
Count	Analog	Indicates the number of publishers configured for the channel.

Channel[] Feedbacks - Publisher[n] (n = 1 to 5):*

Name	Type	Description
IsAvailable	Digital	Indicates a publisher at index <i>n</i> is configured and available to use.
StreamingType	String	Displays the streaming type of publisher (n) on the specified channel as a user-friendly name.
StreamingType.Feedback	Analog	Specifies the streaming type of publisher (n) on the specified channel, where: 0 = Unknown 1 = RTSP 2 = RTMP 3 = WowzaStreamingCloud 4 = OriginalLiveStream 5 = RTP_UDP_Push 6 = MPEG_TS_UDP 7 = MPEG_TS_RTP
StreamingState	String	Displays the streaming state of publisher (n) on the specified channel as a user-friendly description.
StreamingState.Feedback	Analog	Indicates the publishers state as an analog value, where: Unknown/Publisher not configured = 0 Starting = 1 Listening = 2 Started = 3

		Stopped =4 Error = 5
StreamName	String	Displays the user-configurable name for publisher(n) of the specified channel
StreamDuration	String	Displays the streaming duration time of publisher (n) on the specified channel in hh:mm:ss format.

Channel[] Input - Layout*

Name	Type	Description
Select	Analog	Changes the active layout on the specified channel. The first layout in the list is layout number 1.

Channel[] Inputs - Layout[n] (n = 1 to 10):*

Name	Type	Description
Select	Digital	Changes the active layout for the specified channel on a rising edge.
Preview.Enable	Digital	Latch high to enable retrieving preview images for the specified layout for the channel from the Pearl device. Preview images are retrieved every 5 seconds from the device for each preview image item enabled. Generating preview images is a computationally expensive for the Epiphan Pearl device. Epiphan recommends minimizing the amount of preview images being generated where possible. It is recommended to review the CPU load metrics indicated by the Epiphan Pearl webpage during a “worst case” scenario while commissioning the system to determine if the device is still operating within its limits during this scenario. For more details about preview images, consult the “Technical Tips” section at the end of this document.

Channel[] Feedbacks - Layout:*

Name	Type	Description
Count	Analog	Indicates the number of layouts configured for the specified channel.
Selected.Feedback	Analog	Indicates the active layout on the channel. The first layout in the list is layout number 1.

Channel[] Feedbacks - Layout[n] (n = 1 to 10):*

Name	Type	Description
Selected.Feedback	Digital	A high value indicates that the layout on the specified channel is active.

PreviewUrl	String	The preview URL string of the layout on the specified channel. This is provided in a format that is not a complete URL, rather it describes the path of the resource for the preview image on the Crestron processor webserver that the module is running on. See the Technical Tips section towards the end of this document that describes how the processor must be configured and how to build the full URL to pass to the user interface.
Name	String	Displays the user-configured name of the active layout for the specified channel.

Channel[] Feedback - AudioLevel:*

Name	Type	Description
Left	Analog	The left audio channel level for the specified Epiphan Pearl channel. Values 0-100. The values 0-100 are mapped to -60 to 0 dB internally in a linear fashion.
Right	Analog	The left audio channel level for the specified Epiphan Pearl channel. Values 0-100. The values 0-100 are mapped to -60 to 0 dB internally in a linear fashion.

6 Inputs

An Epiphan Pearl has a varying number of inputs depending on the model and its configuration. Although the *physical* inputs such as an SDI, HDMI, or XLR cables connected to the device are fixed in number, the device can be configured for a number of *network* inputs received directly from the AV network, such as NDI, RTSP, SRT, etc streaming media. Inputs that scrape a web page are also available (the “web graphics” term describes these inputs). To account for a wide variety of use cases, the module supports 25 inputs in total regardless of their type.

The number of *network* inputs supported by each device is not unlimited. Please consult other documentation provided by Epiphan for limitations on the number of network streams that can be configured for a device.

As before, should your application have a need for more inputs on the module than 25, please contact Epiphan with a feature request about having support for more added.

Input[] Inputs – 1 to 25:*

Name	Type	Description
Preview.Enable	Digital	Latch high to enable retrieving preview images for the specified input from the Pearl device. Preview images are retrieved every 5 seconds from the device for each preview image item enabled. Generating preview images is a computationally expensive for the Epiphan Pearl device. Epiphan recommends minimizing the amount of preview images being generated where possible. It is recommended to review the CPU load metrics indicated by the Epiphan Pearl webpage during a “worst case” scenario while commissioning the system to determine if the device is still operating within its limits during this scenario. For more details about preview images, consult the “Technical Tips” section at the end of this document.

Feedback:

Name	Type	Description
Count.Feedback	Analog	The total number of inputs on the device. A sum of all of the physical inputs present (including audio-only inputs) and the configured network inputs. (Max Size = 25)

Input[*] Feedback – 1 to 25:

Name	Type	Description
IsAvailable	Digital	A high value indicates an input at the index is available on the module.
Name	String	Displays the user-configurable name for the input.
Type	String	Indicates the input type as a user-friendly name. Values used are "None", "Embedded", "Usb", "Rtsp", "Ndi", "Srt", "WebGraphics".
Type.Feedback	Analog	Indicates the type of input for the input. Values: 0 – "None" (no input present at this index) 1 – "Embedded". All physical inputs (HDMI, SDI, XLR audio, etc) on the device are indicated as this type 2 – USB 3 - RTSP 4 – NDI 5 – SRT 6 – Web Graphics
PreviewUrl	String	The preview URL string for the input. This is provided in a format that is not a complete URL, rather it describes the path of the resource for the preview image on the Crestron processor webserver that the module is running on. See the Technical Tips section towards the end of this document that describes how the processor must be configured and how to build the full URL to pass to the user interface.
IsAudio.Feedback	Digital	A high value indicates the input supports audio. For streaming media or physical connectors where audio is configurable or negotiated (such as with HDMI), this value will be high if the input <i>can support</i> audio, not necessarily if audio is currently present/transmitting. Audio-only inputs will have this value as high but correspondingly <i>IsVideo.Feedback</i> will be low.

IsVideo.Feedback	Digital	A high value indicates the input supports video.
HasSignal.Feedback	Digital	A high value indicates the input is receiving a valid signal. For HDMI/SDI inputs, this indicates a valid video signal is being transmitted from the source. For network inputs, it indicates a stream is actively being received over the network. Audio-only inputs are indicated with a high value on this signal if an audio signal is valid on the connecting cable above a threshold determined by the Epiphan Pearl itself.
HasSignal.Feedback*	Digital	Complement of " <i>HasSignal.Feedback</i> " to indicate that specified input does not have a signal
Resolution	String	Displays the video resolution value of the specified input provided by the Epiphan Pearl. When no active source is connected to the input is connected the value of this signal is "No Signal". Audio-only inputs with no associated resolution will cause the value of this signal to be "active" when <i>HasSignal.Feedback</i> is high, and "No Signal" otherwise. If there is signal present but the module cannot obtain resolution information, a value of "N/A" is used for this signal.
AudioLevel.Left	Analog	The left audio channel level for the specified Epiphan Pearl input. Values 0-100. The values 0-100 are mapped to -60 to 0 dB internally in a linear fashion.
AudioLevel.Right	Analog	The left audio channel level for the specified Epiphan Pearl input. Values 0-100. The values 0-100 are mapped to -60 to 0 dB internally in a linear fashion.

7 Outputs

Output[] Inputs - 1 to 5:*

Name	Type	Description
Preview.Enable	Toggle	Latch high to enable retrieving preview images for the specified output from the Pearl device. Preview images are retrieved every 5 seconds from the device for each preview image item enabled. Generating preview images is a computationally expensive for the Epiphan Pearl device. Epiphan recommends minimizing the amount of preview images being generated where possible. It is recommended to review the CPU load metrics indicated by the Epiphan Pearl webpage during a “worst case” scenario while commissioning the system to determine if the device is still operating within its limits during this scenario. For more details about preview images, consult the “Technical Tips” section at the end of this document.

Feedback:

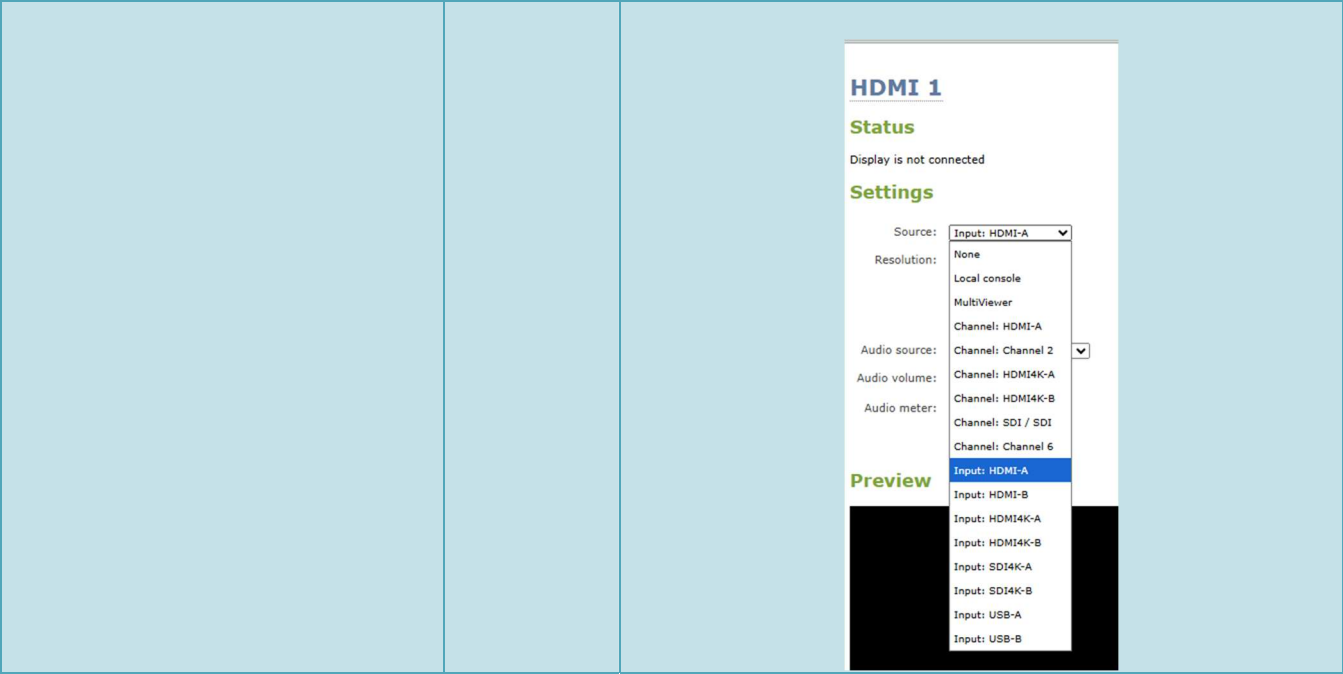
Name	Type	Description
Count.Feedback	Analog	The number of total outputs present on the device. (Max Size = 25)

Output[] Feedbacks – 1 to 5:*

Name	Type	Description
IsAvailable	Digital	A high value specifies if the output at the corresponding index on the module is available.
Name	String	Indicates the user-configurable friendly name of the specified output.
Type	String	Indicates the type of output as a friendly text value. Note that at the current time, only Epiphan Pearl devices with HDMI outputs are available. Until future

		devices offer other output formats, this value will always be "HDMI".
Type.Feedback	Analog	Displays the output type, where: 1 = HDMI Note that at the current time, only Epiphan Pearl devices with HDMI outputs are available. Until future devices offer other output formats, this value will always be 1.
PreviewUrl	String	The preview URL string for the output. This is provided in a format that is not a complete URL, rather it describes the path of the resource for the preview image on the Crestron processor webserver that the module is running on. See the Technical Tips section towards the end of this document that describes how the processor must be configured and how to build the full URL to pass to the user interface.
HasSignal.Feedback	Digital	A high value indicates the output sink is connected and receiving a valid signal from the Pearl. For HDMI outputs, this indicates a valid video signal is being transmitted to the sink connected on the output.
HasSignal.Feedback*	Digital	Complement of "HasSignal.Feedback" to indicate that specified output does not have a signal
IsAudio.Feedback	Digital	A high value indicates the output supports audio.
IsVideo.Feedback	Digital	A high value indicates the output supports video. Note that at the current time, only Epiphan Pearl devices with HDMI outputs are available. Until future devices offer other output formats, this value will always be high.
Resolution	String	Displays the resolution for the video signal on the specified output. If no sink is present on the output, then the value of this signal will be "Disconnected".
ResolutionAndRate	String	Displays the resolution and rate value of the specified output. If no sink is present on the output, then the value of this signal will be "Disconnected".

Source	Analog	The active source for the specified output. The source numbering/order is the same as what is available on the web UI when selecting the source for the output. The first three are neither channels or inputs, but rather built in: 0 – No source 1 – Local Console 2 – Mutliviewer Following this are the channels configured on the device, and then the inputs available/configured. So, the rest of the source list numbering might look something like: 3 – Channel: Channel A 4 – Channel: Channel B 5 – Input: HDMI-A 6 – Input: HDMI-B 7 – Input: HDMI-4K-A 8 – Input: HDMI-4K-B 9 – Input: SDI-4K-A 10 – Input: SDI-4K-B 11 – Input: USB-A 12 – Input: USB-B
Source.Name	String	Displays the user-configurable friendly name of the channel, input, or item routed to the output. The exact string will be as they are presented in the list of sources for the output on the web UI:



8 CMS Kaltura

Inputs:

Name	Type	Description
CurrentEvent.ExtendBy5Minutes	Digital	Extends the current event by 5 minutes on the rising edge.
CurrentEvent.ExtendBy10Minutes	Digital	Extends the current event by 10 minutes on the rising edge.
CurrentEvent.ExtendBy15Minutes	Digital	Extends the current event by 15 minutes on the rising edge.
CurrentEvent.ExtendDuration	Analog	Specifies the duration, in seconds, to extend the event by when the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal is used.
CurrentEvent.ExtendBySpecifiedDuration	Analog	Extends the event by the number of seconds indicated by the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal on the rising edge.
CurrentEvent.Stop	Digital	Stops the current event on the rising edge
NextEvent.Start	Digital	Starts the next scheduled event on the rising edge Note: the current event should be stopped before the next scheduled event is started. If an event is still running, triggering the start of the next event with this signal will not do anything.
CreateEvent.Username	String	Set the username used to create an ad-hoc event with the Kaltura CMS provider.
CreateEvent.Title	String	Set the title of the ad-hoc event you wish to create
CreateEvent.Duration\$	String	Set the duration of the event to create, as a number of minutes. The input signal must be a whole number greater than zero. If the input cannot be converted to a number, the <i>CreateEvent.Create.Message</i> signal will contain a message explaining this.
CreateEvent.Duration	Analog	Set the duration for the event you wish to create (in minutes)
CreateEvent.SelectLive	Digital	Set the type of event to create as "Live"
CreateEvent.SelectVodLive	Digital	Set the type of event as a "Live" and "VOD" type.

CreateEvent.SelectVod	Digital	Set the type of event to create as "VOD".
CreateEvent.Create.Go	Digital	Create the event based on the above parameters specified by their corresponding signals on a rising edge.

Feedback:

Name	Type	Description
IsAvailable	Digital	When high, indicates that the configured CMS provider on the Pearl device is Kaltura
CurrentEvent.IsAvailable	Digital	When high, indicates that current event information is available
CurrentEvent.Title	String	Indicates the title of the current event
CurrentEvent.StartTime	String	Indicates the start time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.EndTime	String	Indicates the end time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.TimeUntilEventEnds	String	A string that indicates how much time is left in the current event. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
CurrentEvent.CanExtend	Digital	feedback value to indicate that the current event has capabilities to extend its time
NextEvent.IsAvailable	Digital	A high value indicates that information on the next event is available
NextEvent.Title	String	The title of the next scheduled event.
NextEvent.StartTime	String	Indicates the start time of the next scheduled event, as a local date and time.

		The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
NextEvent.TimeUntilEventStarts	String	A string that indicates the time remaining until the next event starts. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
NextEvent.CanStart	Digital	A feedback value of high indicates that the next scheduled event can be started
CreateEvent.Create.Success	Digital	Indicates the last created event using this module was created successfully. Will briefly pulse on success.
CreateEvent.Create.Error	Digital	Indicates the last created event using this module was not created successfully. Will briefly pulse on an error occurring, and the error message indicated by the Pearl (or internally by the module itself) is present on the <i>CreateEvent.Create.Message</i> signal.
CreateEvent.Create.Message	String	The error message generated by the Epiphan Pearl device or by the module internally if the event could not be created successfully.
CreateEvent.SelectLive.Feedback	Digital	A high value indicates that a "Live" event type will be created.
CreateEvent.SelectVodLive.Feedback	Digital	A high value indicates that the event created will be both the "VOD" and "Live" type.
CreateEvent.SelectVod.Feedback	Digital	A high value indicates that a "VOD" event type will be created.

9 CMS Panopto

Inputs:

Name	Type	Description
CurrentEvent.ExtendBy5Minutes	Digital	Extends the current event by 5 minutes on the rising edge.
CurrentEvent.ExtendBy10Minutes	Digital	Extends the current event by 10 minutes on the rising edge.
CurrentEvent.ExtendBy15Minutes	Digital	Extends the current event by 15 minutes on the rising edge.
CurrentEvent.ExtendDuration	Analog	Specifies the duration, in seconds, to extend the event by when the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal is used.
CurrentEvent.ExtendBySpecifiedDuration	Analog	Extends the event by the number of seconds indicated by the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal on the rising edge.
CurrentEvent.Stop	Digital	Stops the current event on the rising edge
CurrentEvent.Play Pause.Togg	Digital	Toggles the play/pause state of the current event on a rising edge
CurrentEvent.Play	Digital	Un-pauses the current event on a rising edge
CurrentEvent.Pause	Digital	Pauses the current event on a rising edge
NextEvent.Start	Digital	Starts the next scheduled event on the rising edge Note: the current event should be stopped before the next scheduled event is started. If an event is still running, triggering the start of the next event with this signal will not do anything.
CreateEvent.Title	String	Set the title of the ad-hoc event you wish to create
CreateEvent.Duration\$	String	Set the duration of the event to create, as a number of minutes. The input signal must be a whole number greater than zero. If the input cannot be converted to a number, the <i>CreateEvent.Create.Message</i> signal will contain a message explaining this.
CreateEvent.Duration	Analog	Set the duration for the event you wish to create (in minutes)

CreateEvent.SelectLive	Digital	Set the type of event to create as “Live”
CreateEvent.SelectVod	Digital	Set the type of event to create as “VOD”.
CreateEvent.Create.Go	Digital	Create the event based on the above parameters specified by their corresponding signals on a rising edge.

Feedback:

Name	Type	Description
IsAvailable	Digital	When high, indicates that the configured CMS provider on the Pearl device is Panopto
CurrentEvent.IsAvailable	Digital	When high, indicates that current event information is available
CurrentEvent.Title	String	Indicates the title of the current event
CurrentEvent.StartTime	String	Indicates the start time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.EndTime	String	Indicates the end time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.TimeUntilEventEnds	String	A string that indicates how much time is left in the current event. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be “1 day 12:34:56” to describe a timespan of 36 hours, 34 minutes and 56 seconds.
CurrentEvent.IsRunning	Digital	A feedback value to indicate that the current event is running
CurrentEvent.IsPaused	Digital	A feedback value to indicate that the current event is paused
CurrentEvent.CanExtend	Digital	A feedback value to indicate that the current event has capabilities to extend its time
NextEvent.IsAvailable	Digital	A high value indicates that information on the next scheduled event is available

NextEvent.Title	String	The title of the next scheduled event.
NextEvent.StartTime	String	Indicates the start time of the next scheduled event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
NextEvent.TimeUntilEventStarts	String	A string that indicates the time remaining until the next event starts. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
NextEvent.CanStart	Digital	A feedback value of high indicates that the next scheduled event can be started
CreateEvent.Create.Success	Digital	Indicates the last created event using this module was created successfully. Will briefly pulse on success.
CreateEvent.Create.Error	Digital	Indicates the last created event using this module was not created successfully. Will briefly pulse on an error occurring, and the error message indicated by the Pearl (or internally by the module itself) is present on the <i>CreateEvent.Create.Message</i> signal.
CreateEvent.Create.Message	String	The error message generated by the Epiphan Pearl device or by the module internally if the event could not be created successfully.
CreateEvent.SelectLive.Feedback	Digital	A high value indicates that a "Live" event type will be created.
CreateEvent.SelectVod.Feedback	Digital	A high value indicates that a "VOD" event type will be created.

10 CMS YuJa

Inputs:

Name	Type	Description
CurrentEvent.ExtendBy5Minutes	Digital	Extends the current event by 5 minutes on the rising edge.
CurrentEvent.ExtendBy10Minutes	Digital	Extends the current event by 10 minutes on the rising edge.
CurrentEvent.ExtendBy15Minutes	Digital	Extends the current event by 15 minutes on the rising edge.
CurrentEvent.ExtendDuration	Analog	Specifies the duration, in seconds, to extend the event by when the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal is used.
CurrentEvent.ExtendBySpecifiedDuration	Analog	Extends the event by the number of seconds indicated by the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal on the rising edge.
CurrentEvent.Stop	Digital	Stops the current event on the rising edge
CurrentEvent.Play Pause.Togg	Digital	Toggles the play/pause state of the current event on a rising edge
CurrentEvent.Play	Digital	Un-pauses the current event on a rising edge
CurrentEvent.Pause	Digital	Pauses the current event on a rising edge
NextEvent.Start	Digital	Starts the next scheduled event on the rising edge Note: the current event should be stopped before the next scheduled event is started. If an event is still running, triggering the start of the next event with this signal will not do anything.

Feedback:

Name	Type	Description
IsAvailable	Digital	When high, indicates that the configured CMS provider on the Pearl device is YuJa
CurrentEvent.IsAvailable	Digital	When high, indicates that current event information is available
CurrentEvent.Title	String	Indicates the title of the current event

CurrentEvent.StartTime	String	Indicates the start time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.EndTime	String	Indicates the end time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.TimeUntilEventEnds	String	A string that indicates how much time is left in the current event. The format is HH:mm:ss. If the duration is greater than 24 hours, than the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
CurrentEvent.IsRunning	Digital	A feedback value to indicate that the current event is running
CurrentEvent.IsPaused	Digital	A feedback value to indicate that the current event is paused
CurrentEvent.CanExtend	Digital	A feedback value to indicate that the current event has capabilities to extend its time
NextEvent.IsAvailable	Digital	A high value indicates that information on the next event is available
NextEvent.Title	String	The title of the next scheduled event.
NextEvent.StartTime	String	Indicates the start time of the next scheduled event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
NextEvent.TimeUntilEventStarts	String	A string that indicates the time remaining until the next event starts. The format is HH:mm:ss. If the duration is greater than 24 hours, than the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
NextEvent.CanStart	Digital	A feedback value of high indicates that the next scheduled event can be started

11 CMS Opencast

Inputs:

Name	Type	Description
CurrentEvent.ExtendBy5Minutes	Digital	Extends the current event by 5 minutes on the rising edge.
CurrentEvent.ExtendBy10Minutes	Digital	Extends the current event by 10 minutes on the rising edge.
CurrentEvent.ExtendBy15Minutes	Digital	Extends the current event by 15 minutes on the rising edge.
CurrentEvent.ExtendDuration	Analog	Specifies the duration, in seconds, to extend the event by when the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal is used.
CurrentEvent.ExtendBySpecifiedDuration	Analog	Extends the event by the number of seconds indicated by the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal on the rising edge.
CurrentEvent.Stop	Digital	Stops the current event on the rising edge
CurrentEvent.Play Pause.Togg	Digital	Toggles the play/pause state of the current event on a rising edge
CurrentEvent.Play	Digital	Un-pauses the current event on a rising edge
CurrentEvent.Pause	Digital	Pauses the current event on a rising edge
NextEvent.Start	Digital	Starts the next scheduled event on the rising edge Note: the current event should be stopped before the next scheduled event is started. If an event is still running, triggering the start of the next event with this signal will not do anything.
CreateEvent.Title	String	Set the title of the ad-hoc event you wish to create
CreateEvent.Duration\$	String	Set the duration of the event to create, as a number of minutes. The input signal must be a whole number greater than zero. If the input cannot be converted to a number, the <i>CreateEvent.Create.Message</i> signal will contain a message explaining this.
CreateEvent.Duration	Analog	Set the duration for the event you wish to create (in minutes)

CreateEvent.Create.Go	Digital	Create the event based on the above parameters specified by their corresponding signals on a rising edge.
-----------------------	---------	---

Feedback:

Name	Type	Description
IsAvailable	Digital	When high, indicates that the configured CMS provider on the Pearl device is Opencast
CurrentEvent.IsAvailable	Digital	When high, indicates that current event information is available
CurrentEvent.Title	String	Indicates the title of the current event
CurrentEvent.StartTime	String	Indicates the start time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.EndTime	String	Indicates the end time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.TimeUntilEventEnds	String	A string that indicates how much time is left in the current event. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
CurrentEvent.IsRunning	Digital	A feedback value to indicate that the current event is running
CurrentEvent.IsPaused	Digital	A feedback value to indicate that the current event is paused
CurrentEvent.CanExtend	Digital	A feedback value to indicate that the current event has capabilities to extend its time
NextEvent.IsAvailable	Digital	A high value indicates that information on the next event is available
NextEvent.Title	String	The title of the next scheduled event.

NextEvent.StartTime	String	Indicates the start time of the next scheduled event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
NextEvent.TimeUntilEventStarts	String	A string that indicates the time remaining until the next event starts. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be “1 day 12:34:56” to describe a timespan of 36 hours, 34 minutes and 56 seconds.
NextEvent.CanStart	Digital	A feedback value of high indicates that the next scheduled event can be started
CreateEvent.Create.Success	Digital	Indicates the last created event using this module was created successfully. Will briefly pulse on success.
CreateEvent.Create.Error	Digital	Indicates the last created event using this module was not created successfully. Will briefly pulse on an error occurring, and the error message indicated by the Pearl (or internally by the module itself) is present on the <i>CreateEvent.Create.Message</i> signal.
CreateEvent.Create.Message	String	The error message generated by the Epiphan Pearl device or by the module internally if the event could not be created successfully.

12 CMS Epiphan Edge

Inputs:

Name	Type	Description
CurrentEvent.Stop	Digital	Stops the current event on the rising edge
NextEvent.Start	Digital	Starts the next scheduled event on the rising edge Note: the current event should be stopped before the next scheduled event is started. If an event is still running, triggering the start of the next event with this signal will not do anything.

Feedback:

Name	Type	Description
IsAvailable	Digital	When high, indicates that the configured CMS provider on the Pearl device is Epiphan Edge
CurrentEvent.IsAvailable	Digital	When high, indicates that current event information is available
CurrentEvent.Title	String	Indicates the title of the current event
CurrentEvent.StartTime	String	Indicates the start time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.EndTime	String	Indicates the end time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.TimeUntilEventEnds	String	A string that indicates how much time is left in the current event. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.

NextEvent.IsAvailable	Digital	A high value indicates that information on the next scheduled event is available
NextEvent.Title	String	The title of the next scheduled event.
NextEvent.StartTime	String	Indicates the start time of the next scheduled event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
NextEvent.TimeUntilEventStarts	String	A string that indicates the time remaining until the next event starts. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
NextEvent.CanStart	Digital	A feedback value of high indicates that the next scheduled event can be started
NextEvent.IsAvailable	Digital	A high value indicates that information on the next event is available

13 CMS Echo360

Inputs:

Name	Type	Description
CurrentEvent.ExtendBy5Minutes	Digital	Extends the current event by 5 minutes on the rising edge.
CurrentEvent.ExtendBy10Minutes	Digital	Extends the current event by 10 minutes on the rising edge.
CurrentEvent.ExtendBy15Minutes	Digital	Extends the current event by 15 minutes on the rising edge.
CurrentEvent.ExtendDuration	Analog	Specifies the duration, in seconds, to extend the event by when the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal is used.
CurrentEvent.ExtendBySpecifiedDuration	Analog	Extends the event by the number of seconds indicated by the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal on the rising edge.
CurrentEvent.Stop	Digital	Stops the current event on the rising edge
CurrentEvent.Play Pause.Togg	Digital	Toggles the play/pause state of the current event on a rising edge
CurrentEvent.Play	Digital	Un-pauses the current event on a rising edge
CurrentEvent.Pause	Digital	Pauses the current event on a rising edge
NextEvent.Start	Digital	Starts the next scheduled event on the rising edge Note: the current event should be stopped before the next scheduled event is started. If an event is still running, triggering the start of the next event with this signal will not do anything.
CreateEvent.Title	String	Set the title of the ad-hoc event you wish to create
CreateEvent.Duration\$	String	Set the duration of the event to create, as a number of minutes. The input signal must be a whole number greater than zero. If the input cannot be converted to a number, the <i>CreateEvent.Create.Message</i> signal will contain a message explaining this.
CreateEvent.Duration	Analog	Set the duration for the event you wish to create (in minutes)

CreateEvent.SelectLive	Digital	Set the type of event to create as “Live”
CreateEvent.SelectVod	Digital	Set the type of event to create as “VOD”.
CreateEvent.Create.Go	Digital	Create the event based on the above parameters specified by their corresponding signals on a rising edge.

Feedback:

Name	Type	Description
IsAvailable	Digital	When high, indicates that the configured CMS provider on the Pearl device is Echo360.
CurrentEvent.IsAvailable	Digital	When high, indicates that current event information is available
CurrentEvent.Title	String	Indicates the title of the current event
CurrentEvent.StartTime	String	Indicates the start time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.EndTime	String	Indicates the end time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.TimeUntilEventEnds	String	A string that indicates how much time is left in the current event. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be “1 day 12:34:56” to describe a timespan of 36 hours, 34 minutes and 56 seconds.
CurrentEvent.IsRunning	Digital	A feedback value to indicate that the current event is running
CurrentEvent.IsPaused	Digital	A feedback value to indicate that the current event is paused
CurrentEvent.CanExtend	Digital	A feedback value to indicate that the current event has capabilities to extend its time
NextEvent.IsAvailable	Digital	A high value indicates that information on the next scheduled event is available

NextEvent.Title	String	The title of the next scheduled event.
NextEvent.StartTime	String	Indicates the start time of the next scheduled event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
NextEvent.TimeUntilEventStarts	String	A string that indicates the time remaining until the next event starts. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
NextEvent.CanStart	Digital	A feedback value of high indicates that the next scheduled event can be started
CreateEvent.Create.Success	Digital	Indicates the last created event using this module was created successfully. Will briefly pulse on success.
CreateEvent.Create.Error	Digital	Indicates the last created event using this module was not created successfully. Will briefly pulse on an error occurring, and the error message indicated by the Pearl (or internally by the module itself) is present on the <i>CreateEvent.Create.Message</i> signal.
CreateEvent.Create.Message	String	The error message generated by the Epiphan Pearl device or by the module internally if the event could not be created successfully.
CreateEvent.SelectLive.Feedback	Digital	A high value indicates that a "Live" event type will be created.
CreateEvent.SelectVod.Feedback	Digital	A high value indicates that a "VOD" event type will be created.

14 CMS Generic Container

As of firmware 4.24.6, Epiphan added Echo360 as a CMS provider. It runs in a “container” and thus supports a common CMS feature set that any other CMS provider wishing to use the “containerized” CMS system on the Pearl. To support that future need, the “generic container” CMS type was added to the module as well, which supports all “containerized” CMS system types.

Inputs:

Name	Type	Description
CurrentEvent.ExtendBy5Minutes	Digital	Extends the current event by 5 minutes on the rising edge.
CurrentEvent.ExtendBy10Minutes	Digital	Extends the current event by 10 minutes on the rising edge.
CurrentEvent.ExtendBy15Minutes	Digital	Extends the current event by 15 minutes on the rising edge.
CurrentEvent.ExtendDuration	Analog	Specifies the duration, in seconds, to extend the event by when the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal is used.
CurrentEvent.ExtendBySpecifiedDuration	Analog	Extends the event by the number of seconds indicated by the <i>CurrentEvent.ExtendBySpecifiedDuration</i> signal on the rising edge.
CurrentEvent.Stop	Digital	Stops the current event on the rising edge
CurrentEvent.Play Pause.Togg	Digital	Toggles the play/pause state of the current event on a rising edge
CurrentEvent.Play	Digital	Un-pauses the current event on a rising edge
CurrentEvent.Pause	Digital	Pauses the current event on a rising edge
NextEvent.Start	Digital	Starts the next scheduled event on the rising edge Note: the current event should be stopped before the next scheduled event is started. If an event is still running, triggering the start of the next event with this signal will not do anything.
CreateEvent.Title	String	Set the title of the ad-hoc event you wish to create
CreateEvent.Duration\$	String	Set the duration of the event to create, as a number of minutes.

		The input signal must be a whole number greater than zero. If the input cannot be converted to a number, the <i>CreateEvent.Create.Message</i> signal will contain a message explaining this.
CreateEvent.Duration	Analog	Set the duration for the event you wish to create (in minutes)
CreateEvent.SelectLive	Digital	Set the type of event to create as "Live"
CreateEvent.SelectVod	Digital	Set the type of event to create as "VOD".
CreateEvent.Create.Go	Digital	Create the event based on the above parameters specified by their corresponding signals on a rising edge.

Feedback:

Name	Type	Description
IsAvailable	Digital	When high, indicates that the configured CMS provider on the Pearl device is Panopto
CurrentEvent.IsAvailable	Digital	When high, indicates that current event information is available
CurrentEvent.Title	String	Indicates the title of the current event
CurrentEvent.StartTime	String	Indicates the start time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.EndTime	String	Indicates the end time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
CurrentEvent.TimeUntilEventEnds	String	A string that indicates how much time is left in the current event. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
CurrentEvent.IsRunning	Digital	A feedback value to indicate that the current event is running

CurrentEvent.IsPaused	Digital	A feedback value to indicate that the current event is paused
CurrentEvent.CanExtend	Digital	A feedback value to indicate that the current event has capabilities to extend its time
NextEvent.IsAvailable	Digital	A high value indicates that information on the next scheduled event is available
NextEvent.Title	String	The title of the next scheduled event.
NextEvent.StartTime	String	Indicates the start time of the next scheduled event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
NextEvent.TimeUntilEventStarts	String	A string that indicates the time remaining until the next event starts. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
NextEvent.CanStart	Digital	A feedback value of high indicates that the next scheduled event can be started
CreateEvent.Create.Success	Digital	Indicates the last created event using this module was created successfully. Will briefly pulse on success.
CreateEvent.Create.Error	Digital	Indicates the last created event using this module was not created successfully. Will briefly pulse on an error occurring, and the error message indicated by the Pearl (or internally by the module itself) is present on the <i>CreateEvent.Create.Message</i> signal.
CreateEvent.Create.Message	String	The error message generated by the Epiphan Pearl device or by the module internally if the event could not be created successfully.
CreateEvent.SelectLive.Feedback	Digital	A high value indicates that a "Live" event type will be created.
CreateEvent.SelectVod.Feedback	Digital	A high value indicates that a "VOD" event type will be created.

15 Events

Feedback:

Name	Type	Description
Cms.Provider.Type.Feedback	Analog	The analog value corresponding to the configured CMS provider on the device, where: 0 = Unknown (provider yet to be determined) 1 = None – No provider is configured 2 = Kaltura 3 = Panopto 4 = Opencast 5= YuJa 6= Epiphan Edge
Cms.Enabled.Feedback	Digital	If high, indicates that a CMS provider is configured and event/CMS functions are available

Event.Upcoming - Inputs:

Name	Type	Description
Start	Digital	Starts the next scheduled event on the rising edge Note: the current event should be stopped before the next scheduled event is started. If an event is still running, triggering the start of the next event with this signal will not do anything.

Event.Upcoming - Feedback:

Name	Type	Description
CanStart	Digital	A feedback value of high indicates that the next scheduled event can be started
Title	String	The title of the next scheduled event.
StartTime	String	Indicates the start time of the next scheduled event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM

EndTime	String	Indicates the end time of the next scheduled event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
Status	String	Indicates the status of the next scheduled event (Scheduled, Running, Paused, Finished)
Tags	String	Displays the tags tied to the upcoming event
TimeUntilEventStarts	String	A string that indicates the time remaining until the next event starts. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
Status.Feedback	Analog	A numerical value to indicate the next scheduled event status, where: 0 = None 1 = Scheduled 2 = Running 3 = Paused 4 = Finished 5 = Skipped

Event.Current - Inputs:

Name	Type	Description
Stop	Digital	Stops the current event on the rising edge
Resume	Digital	Un-pauses the current event on a rising edge
Pause	Digital	Pauses the current event on a rising edge
ExtendBy5Minutes	Digital	Extends the current event by 5 minutes on the rising edge.
ExtendBy10Minutes	Digital	Extends the current event by 10 minutes on the rising edge.
ExtendBy15Minutes	Digital	Extends the current event by 15 minutes on the rising edge.
ExtendDuration	Analog	Specifies the duration, in seconds, to extend the event by when the <i>ExtendBySpecifiedDuration</i> signal is used.

ExtendBySpecifiedDuration	Analog	Extends the event by the number of seconds indicated by the <i>ExtendBySpecifiedDuration</i> signal on the rising edge.
---------------------------	--------	---

Event.Current - Feedback:

Name	Type	Description
Stopped.Feedback	Digital	Indicator to set the current event has stopped
Is.Running	Digital	Indicator to set the current indicator is running
Is.Paused	Digital	Indicator to set the current indicator has paused
CanStop	Digital	A feedback value to indicate that the current event has capabilities to stop the current event
CanPause	Digital	A feedback value to indicate that the current event has capabilities to pause the current event
CanExtend	Digital	A feedback value to indicate that the current event has capabilities to extend the current event
CanResume	Digital	A feedback value to indicate that the current event has capabilities to resume the current event
Title	String	Indicates the title of the current event
StartTime	String	Indicates the start time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
EndTime	String	Indicates the end time of the current event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
Status	String	Indicates the status of the current event (Scheduled, Running, Paused, Finished)
Tags	String	Displays the tags tied to the current event
TimeUntilEventEnds	String	A string that indicates how much time is left in the current event. The format is HH:mm:ss. If the duration is greater than 24 hours, than the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.

Status.Feedback	Analog	A numerical value to represent the event status, where: 0 = None 1 = Scheduled 2 = Running 3 = Paused 4 = Finished 5 = Skipped
-----------------	--------	--

Event.Create - Inputs:

Name	Type	Description
Title	String	Set the title of the ad-hoc event you wish to create
Username	String	Set the username for the CMS provider related to the event you wish to create
Password	String	Set the username for the CMS provider related to the event you wish to create
Folder	String	Set the folder used for the ad-hoc event you wish to create
Duration\$	String	Set the duration of the event to create, as a number of minutes. The input signal must be a whole number greater than zero. If the input cannot be converted to a number, the <i>CreateEvent.Create.Message</i> signal will contain a message explaining this.
Duration	Analog	Set the duration for the event you wish to create (in minutes)
Type	Analog	Set the type of ad-hoc event to create, where: 1 = Vod 2 = Live 3 = VodLive
Event.Create	Digital	Create the event based on the above parameters specified by their corresponding signals on a rising edge.

Event.Create - Feedback:

Name	Type	Description
------	------	-------------

Success	Digital	Indicates the last created event using this module was created successfully. Will briefly pulse on success.
Error	Digital	Indicates the last created event using this module was not created successfully. Will briefly pulse on an error occurring, and the error message indicated by the Pearl (or internally by the module itself) is present on the <i>CreateEvent.Create.Message</i> signal.
ErrorMessage	String	The error message generated by the Epiphan Pearl device or by the module internally if the event could not be created successfully.

ScheduledEvent[] Inputs – 1 to 10:*

Name	Type	Description
Start	Digital	Start the specified scheduled event.

ScheduledEvent Feedback:

Name	Type	Description
Count.Feedback	Analog	Indicates the number of scheduled events provided by the module Currently, this is the count of scheduled events provided by the Pearl API. Should you need a way to filter the scheduled events for the list of scheduled events provided by the module, please contact Epiphan with the appropriate feature request.

ScheduledEvent[] Feedback – 1 to 10:*

Name	Type	Description
Title	String	Title of the specified event
StartTime	String	Indicates the start time of the specified event, as a local date and time.

		The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
EndTime	String	Indicates the end time of the specified event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
Status	String	Displays the output status of the specified event (Scheduled, Running, Paused, Finished)
Tags	String	Displays the tags tied to the specified event
Status.Feedback	Analog	A numerical value to represent the specified event's status, where: 0 = None 1 = Scheduled 2 = Running 3 = Paused 4 = Finished 5 = Skipped

PastEvent Feedback:

Name	Type	Description
Count.Feedback	Analog	The number of past events today. Past events are all events that have been completed. The "current" event is not included in the list of past events.

PastEvent[] Feedback – 1 to 10:*

Name	Type	Description
Title	String	Title of the specified event
StartTime	String	Indicates the start time of the specified event, as a local date and time. The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
EndTime	String	Indicates the end time of the specified event, as a local date and time.

		The format is MM/dd/yyyy hh:mm:ss AM PM, ie 5/1/2021 6:32:06 PM
Status	String	Displays the output status of the specified event (Scheduled, Running, Paused, Finished)
Tags	String	Displays the tags tied to the specified event
Status.Feedback	Analog	A numerical value to represent the specified event's status, where: 0 = None 1 = Scheduled 2 = Running 3 = Paused 4 = Finished 5 = Skipped

16 Presets

Inputs:

Name	Type	Description
[*]	Digital	Triggers the corresponding device preset on the rising edge to trigger the desired preset. The first preset is preset number 1.
Recall	Analog	Set an analog value to recall a specified preset. The first preset is preset number 1. Note: Recalling a preset may reboot the Pearl device, depending on the configured scope of the preset. Confirm this during the commissioning stage of the project to avoid potential disruptions or negative UX.

Feedback:

Name	Type	Description
Count.Feedback	Analog	A value indicating the number of device presets configured.
Recall.WillReboot	Digital	Pulsed if the recently recalled preset will reboot the Epiphan Pearl.
Recall.Success	Digital	Pulsed if the recently recalled preset was recalled successfully
Recall.Error	Digital	Pulsed if the recently recalled preset could not be recalled.
Recall.Message	String	In the event there was an error recalling the preset, an associated error message. Otherwise a status message related to the recently called preset.
Feedback[*]	Digital	A high to indicates that this preset was the last active preset triggered (1 to 10)
Recalled.Feedback	Analog	The value of the last preset recalled
[*].Name	String	The display name of preset (1 to 10)
[*].Description	String	The description text of preset (1 to 10)

17 System

Inputs:

Name	Type	Description
Reboot	Digital	Reboots the device on a rising edge
FactoryReset	Digital	Factory resets the device on a rising edge
Shutdown	Digital	Shuts down the device on a rising edge.

Feedback:

Name	Type	Description
Uptime	String	Displays how long the device has been online. The format is HH:mm:ss. If the duration is greater than 24 hours, then the amount of days are included, and the format for example would be "1 day 12:34:56" to describe a timespan of 36 hours, 34 minutes and 56 seconds.
High.Feedback	Digital	Indicates that the CPU usage for the Pearl is above a threshold deemed by Epiphan as "high"
CpuLoad	String	Displays the current CPU load as a percentage.
CpuTemp	String	Displays the current temperature of the CPU, in degrees Celsius
CpuTemp.Threshold	String	Displays the threshold temperature of the CPU above which Epiphan deems high.

System.Disk.Internal Feedback:

Name	Type	Description
FreeSpace	String	Displays the free disk space available in the device's internal storage as Mebibytes (2^{20} bytes or 1,024 kibibytes). Ie. "280678 MiB"
FreeSpace.Percent	Analog	Displays the percentage of free space from the total in the device's internal storage of the total as analog value from 0 to 65535 (0% to 100%).
UsedSpace	String	Displays the free disk space available in the device's internal storage as Mebibytes (2^{20} bytes or 1,024 kibibytes). Ie. "280678 MiB"
UsedSpace.Percent	Analog	Displays the percentage of used space from the total in the device's internal storage of the total as analog value from 0 to 65535 (0% to 100%).
TotalSpace	String	Displays the total disk space available in the device's internal storage as Mebibytes (2^{20} bytes or 1,024 kibibytes). Ie. "280678 MiB"
UsagePercent	String	Displays the percentage of used space on the device's internal storage as a string. ie. "67%"

System.Disk.External Inputs:

Name	Type	Description
SafelyEject	Digital	Safely ejects the SD card from the device on the rising edge.

System.Disk.External Feedback:

Name	Type	Description
Connected.Feedback	Digital	A high value indicates external storage is present
State.Feedback	String	Indicates the current state of the external storage as a user-friendly string
State	Analog	Analog feedback value of the current state, where: 0 = None (not yet determined or external storage not supported) 1 = Ready 2 = No Device 3 = Device Detected 4 = Device Detected, Read Only 5 = Device Formatting
FreeSpace	String	Displays the free disk space available in the device's external storage as Mebibytes (2^{20} bytes or 1,024 kibibytes). Ie. "280678 MiB"
FreeSpace.Percent	Analog	Displays the percentage of free space from the total in the device's external storage of the total as analog value from 0 to 65535 (0% to 100%).
UsedSpace	String	Displays the free disk space available in the device's external storage as Mebibytes (2^{20} bytes or 1,024 kibibytes). Ie. "280678 MiB"
UsedSpace.Percent	Analog	Displays the percentage of used space from the total in the device's external storage of the total as analog value from 0 to 65535 (0% to 100%).
TotalSpace	String	Displays the total disk space available in the device's external storage as Mebibytes (2^{20} bytes or 1,024 kibibytes).

		ie. "280678 MiB"
UsagePercent	String	Displays the percentage of used space on the device's external storage as a string. ie. "67%"

18 Device

Feedback:

Name	Type	Description
Name.Feedback	String	Reports back the user-defined device name for the Pearl Device
UserDescription	String	Reports back user defined identity parameter
UserLocation	String	Reports back user defined identity parameter
SerialNumber	String	Reports back the connected device serial number
FirmwareVersion	String	Reports back the connected device firmware version
FirmwareRevision	String	Reports back the connected device firmware revision
Firmware.FullString	String	Concatenation of firmware version and revision
ProductID	String	Reports back the device Product ID
ProductName	String	Reports back the device Product Name

19 Network

The network parameters on the string output signals on the module described below correspond to the primary ethernet port. If a need arises for this information to be available on the module for a USB NIC connected to the Pearl as well, please submit the appropriate feature request to Epiphan.

Feedback:

Name	Type	Description
IP.Address.Feedback	String	Reports back the connected device's IP Address
Netmask.Feedback	String	Reports back the connected device's subnet mask
Gateway.Feedback	String	Reports back the connected device's gateway address
DNS.Feedback	String	Reports back the connected device's DNS address
MAC.Address.Feedback	String	Reports back the connected device's MAC address

20 Debug

Inputs:

Name	Type	Description
Debug_Level	Analog	Indicates the debug level verbosity. There are 5 values that can be used. In order of increasing verbosity: 0 – “Fatal” – Typically, no debug messages except for rare occasions. 1 – “Error” – All the above, and includes debug messages that indicate that something has gone wrong that cannot be handled normally and are impacting normal operation. 2 – “Warning” – All of the above, and includes debug messages that indicate something out of the ordinary or an error that might impact normal operations, but in many cases can be ignored or handled by the module. 3 – “Information” - All of the above, and includes debug messages with information about normal operation. 4 – “Debug” – All of the above, and includes debug messages with information about normal module operation, sequencing, and processing of data to and from the device. 5 – “Verbose” – All available debug messages are included in the output. Importantly, this level includes all REST communications to and from the device. Be prepared to enable/disable logging for a short window or capture the output of the text console to a file for later use. The debug output will be very difficult to follow in real time.
Enable_Debug	Digital	Latch this signal to enable debug messages to be printed to the text console of the Crestron 3- or 4-series appliance, or the error log in the case of a Crestron virtual control application. These debug messages are also available on the <i>Debug.Message</i> signal as described below for use within the SIMPL program.

Feedback:

Name	Type	Description
Debug_Level.Feedback	Analog	Indicates the current debug level setting. Unless set by the programmer using this module, this value defaults to 3, which is the “Information” debug level.
Debug.Message	String	Last reported debug message from the module.

21 Stream Management

The stream management signals can add or modify publishers for each channel.

Inputs – Stream.Management:

Name	Type	Description
Select_Channel	Analog	Selects a channel for the Add_[X]_Stream signals below
Select_Stream	Analog	Selects a stream for the [X]_New_Stream_Settings signals
Add_RTSP_Stream	Digital	Adds a new publisher with an RTSP Stream on the channel specified by the Select_Channel join
Add_RTMP_Stream	Digital	Adds a new publisher with an RTMP Stream on the channel specified by the Select_Channel join
Add_RTPUDP_Stream	Digital	Adds a new publisher with an RTP-UDP Stream on the channel specified by the Select_Channel join
Add_MPEG-TS_RTP_Stream	Digital	Adds a new publisher with an MPEG-TS RTP Stream on the channel specified by the Select_Channel join
Add_MPEG-TS_UDP_Stream	Digital	Adds a new publisher with an MPEG-TS UDP Stream on the channel specified by the Select_Channel join

Inputs – Stream.Management.RTSP:

Name	Type	Description
New_Stream_Settings	Digital	When triggered, modifies the stream settings of the selected channel and stream using the data provided in the RTSP management signals below.
Single_Touch	Digital	Sets the single touch property. Low = false, High = true
Stream_URL	Digital	Sets the stream URL.
Transport	Analog	Sets the transport type. 0 = tcp, 1 = udp, any other value will default to tcp.
Username	String	The stream username.
Password	String	The stream password.

Inputs – Stream.Management.RTMP:

Name	Type	Description
New_Stream_Settings	Digital	When triggered, modifies the stream settings of the selected channel and stream using the data provided in the RTMP management signals below.
Single_Touch	Digital	Sets the single touch property. Low = false, High = true
Stream_URL	Digital	Sets the stream URL.
Stream_Name	String	Sets the stream name.
Username	String	The stream username.
Password	String	The stream password.

Inputs – Stream.Management.RTP_UDP:

Name	Type	Description
New_Stream_Settings	Digital	When triggered, modifies the stream settings of the selected channel and stream using the data provided in the RTMP management signals below.
Address	String	Sets the stream address.
Video_Port	Analog	Sets the stream video port.
Audio_Port	Analog	Sets the stream audio port.

Inputs – Stream.Management.MPEG-TS_UDP:

Name	Type	Description
New_Stream_Settings	Digital	When triggered, modifies the stream settings of the selected channel and stream using the data provided in the MPEG-TS UDP management signals below.
Address	String	Sets the stream address.
Port	Analog	Sets the stream port.
Sap	String	Use SAP announcement. High = true, Low = false.
Sap_IP	String	Sets the SAP IP address.

Sap_Channel	String	Sets the SAP Channel.
Sap_Play_Group	String	Sets the SAP Play Group.

Inputs – Stream.Management.MPEG-TS_RTP:

Name	Type	Description
New_Stream_Settings	Digital	When triggered, modifies the stream settings of the selected channel and stream using the data provided in the MPEG-TS RTP management signals below.
Address	String	Sets the stream address.
Port	Analog	Sets the stream port.
Sap	String	Use SAP announcement. High = true, Low = false.
Sap_IP	String	Sets the SAP IP address.
Sap_Channel	String	Sets the SAP Channel.
Sap_Play_Group	String	Sets the SAP Play Group.

Feedback:

Name	Type	Description
Stream.Management. Error_Message	String	Displays any error messages generated by the stream settings operations.

22 Error

Feedback:

Name	Type	Description
Error.Pulse	Digital	Pulses briefly when a user-friendly error message unrelated to any of the other error message output signals is generated on <i>Error.Message</i> as described below
Error.Message	String	Text indicating the latest user-friendly error message. Use in conjunction with <i>Error.Pulse</i> to indicate to users an error has occurred or to send the error message elsewhere within the program.

Complementary Modules

1 Construct Test Field Debouncer

This module is included in the demo program to work around behavior and bugs with the Crestron construct text field. It is not required to be used in your program and is purely to support the demo program.

Inputs:

Name	Type	Description
Enable_Debug	Digital	Latch this signal to enable console debugging
String.input[*]	String	Text coming in from textfield object

Parameters:

Name	Type	Description
Debounce Time	Time (s)	The delayed time for the unchanged text input to send its value to its respective output.

Feedback:

Name	Type	Description
String.output[*]	String	Return text from input field after debounced

2 Epiphan Pearl Event List Sorting

This module is built to support the demo program. It is not required for use in your program if you don't need it.

The purpose of this helper module is to take the scheduled and past event signal lists provided by the Pearl and concatenate the Title, Start Time, and End time for the scheduled and past events and outputs them as one serial signal in a format suitable for the demo program UI.

Inputs:

Name	Type	Description
Scheduled_Event_Count	Analog	Number of scheduled events coming out of the Pearl module
Past_Event_Count	Analog	Number of past events coming out of the Pearl module
Past_Event_Title[*]	String	Title of the past event (limit = 10)
Past_Event_Start_Time[*]	String	Start time of the past event (limit = 10)
Past_Event_End_Time[*]	String	End time of the past event (limit = 10)
Scheduled_Event_Title[*]	String	Title of today's scheduled event (limit = 10)
Scheduled_Start_Time[*]	String	Start time of today's scheduled event (limit = 10)
Scheduled_End_Time[*]	String	End time of today's scheduled event (limit = 10)

Feedback:

Name	Type	Description
UI_Scheduled_Event_Count	Analog	Adjusts the count of <i>Schedule_Event_Count</i> , to tie into the UI list. If there are no scheduled events, it will create a "No Scheduled Events" entry
UI_Past_Event_Count	Analog	Adjusts the count of <i>Past_Event_Count</i> to tie into the UI list.

		If there are no scheduled events, it will create a “No Past Events” entry.
Scheduled_Event_Text[*]	String	Merges the Title, Start Time and End Time of the scheduled event
Past_Event_Text[*]	String	Merges the Title, Start Time and End Time of the past event

3 Epiphan Pearl Input List Sorting

This helper module supports the demo program by taking the input[x] signals and sorting them into sub-lists for the demo program categorized as video, audio-only, and network inputs. It is not required that you use this module in your own program.

Inputs:

Name	Type	Description
Input_Has_Signal[*]	Digital	From core module of the “ <i>HasSignal.Feedback</i> ” value
Input_Is_Audio[*]	Digital	From core module of the “ <i>isAudio.Feedback</i> ” value
Input_Is_Video[*]	Digital	Native list from core module of the “ <i>isVideo.Feedback</i> ” value
Video_Input_Preview_Enable[*]	Digital	UI trigger to enable image URL preview of physical input source onto UI
Network_Input_Preview_Enable[*]	Digital	UI trigger to enable image URL preview of network input source onto UI
Input_Type_Feedback[*]	Analog	From core module of the “ <i>Type.Feedback</i> ” value
Input_AudioLevel_Left[*]	Analog	From the core module “ <i>AudioLevel.Left</i> ” value
Input_AudioLevel_Right[*]	Analog	From the core module “ <i>AudioLevel.Right</i> ” value
Input_Name[*]	String	From the core module “ <i>Name</i> ” value
Input_Type[*]	String	From the core module “ <i>Type</i> ” value
Input_Preview_Url[*]	String	From the core module “ <i>Preview_Url</i> ” value
Input_Resolution[*]	String	From the core module “ <i>Resolution</i> ” value

Feedback:

Name	Type	Description
Input_Preview_Enabled[*]	Digital	Triggers the <i>"Preview.Enable"</i> value to the core module
Video_Input_Has_Signal[*]	Digital	Filtered out list for physical video inputs having signal
Video_Input_Available[*]	Digital	Filtered out list for physical video inputs available
Video_Input_Is_Audio [*]	Digital	Filtered out list for physical inputs being audio types
Audio_Input_Has_Signal[*]	Digital	Filtered out list for physical audio inputs having signal
Audio_Input_Available[*]	Digital	Filtered out list for physical audio inputs available
Network_Input_Has_Signal[*]	Digital	Filtered out list for network inputs having signal
Network_Input_Available[*]	Digital	Filtered out list for network inputs available
Network_Input_Is_Audio[*]	Digital	Filtered out list for network inputs being audio types
Video_Input_Count_Feedback[*]	Analog	Number of physical video inputs
Audio_Input_Count_Feedback[*]	Analog	Number of physical audio inputs
Network_Input_Count_Feedback[*]	Analog	Number of network inputs
Video_Input_AudioLevel_Left[*]	Analog	Filtered out list for physical video inputs left audio track
Video_Input_AudioLevel_Right[*]	Analog	Filtered out list for physical video inputs right audio track
Audio_Input_AudioLevel_Left[*]	Analog	Filtered out list for physical audio inputs left audio track
Audio_Input_AudioLevel_Right[*]	Analog	Filtered out list for physical audio inputs right audio track
Network_Input_AudioLevel_Left[*]	Analog	Filtered out list for network inputs left audio track
Network_Input_AudioLevel_Right[*]	Analog	Filtered out list for network inputs right audio track

Video_Input_Name[*]	String	Filtered out physical video input names
Video_Input_Type[*]	String	Filtered out physical video input types
Video_Input_Preview_Url[*]	String	Filtered out physical video input's preview URL feed
Video_Input_Resolution[*]	String	Filtered out physical video input's resolution
Audio_Input_Name[*]	String	Filtered out physical audio input names
Audio_Input_Type[*]	String	Filtered out physical audio input types
Audio_Input_Preview_Url[*]	String	Filtered out physical audio input's preview URL feed
Audio_Input_Resolution[*]	String	Filtered out physical audio input's resolution
Network_Input_Name[*]	String	Filtered out network input names
Network_Input_Type[*]	String	Filtered out network input types
Network_Input_Preview_Url[*]	String	Filtered out network input's preview URL feed
Network_Input_Resolution[*]	String	Filtered out network input's resolution

4 Epiphan Pearl Touchpanel Preview URL helper

This helper module is utilized to take the various “preview url” output signals from the module and process them into a full URL that the corresponding touchpanel will be able to access on the Crestron processor's web server. One instance of this helper module is typically needed for each touchpanel IP ID, although if a set of touchpanels are always guaranteed to be on the same network segment, then the outputs from this module can be sent to all touchpanels in that set.

To work around limitations the Crestron Construct image object has and bugs at the time of writing around the touchpanel prompting for image credentials when running a Crestron Construct CH5 UI, the Epiphan Pearl module will fetch the preview image for the inputs, outputs, channels, and channel layouts when enabled and host them internally on the processors own web server. This bypasses the need for image authentication and having to accept the self-signed certificate the images are hosted with when using HTTPS connections. See the Technical Tips section towards the end of this document for a full explanation on configuring the processor properly for preview images.

As the Crestron processor may have multiple network interfaces, the touchpanel may have a different

network address it accesses the processor as part of the full URL when fetching the preview image. This module uses the indicated network address for the touchpanel and matches it up with the appropriate network address for the processor, and then outputs the full, formatted preview URL to send to the image object on your UI to fetch the preview image from the processor.

Using this helper module in conjunction with the core module is not required should you wish to duplicate its functionality in a way that fits your own workflow better, but it is recommended for most scenarios.

Inputs:

Name	Type	Description
Processor.Custom.Address	Serial	An optional signal specifying the processor IP address or hostname to use when formatting the full URL of the preview image on the processor. This module automatically detects the network interfaces on the processor (LAN, Control subnet, USB network interface) and is able to determine the IP address of those interfaces automatically. Should an address be required that is not any of those three, such as if there is a port forward or NAT in use, then this signal must be used.
Touchpanel.Address	Serial	Tie IP address feedback signal of the appropriate device extender for the Crestron touchpanel you are using this module for to this signal input. The module will use this address to attempt to match the IP address scheme with the appropriate address of the LAN, control subnet, or USB network interface when generating the full URL on the output of the module for the corresponding input URL stub. If the <i>Processor.Custom.Address</i> signal is used, this signal is ignored. If the user interface does not have a device extender indicating the touchpanel IP address, (such as with an XPanel), then this signal can be left blank (use the // signal), and the processor IP address to use can be specified directly with the <i>Processor.Custom.Address</i> signal.
PreviewUrl.Unformatted[*]	Serial	These inputs should be tied to the output of the various Preview URL signals on the core Epiphan Pearl module.

Feedback:

Name	Type	Description
Touchpanel.PreviewUrl.Formatted[*]	Signal	This list of outputs is the fully formatted URL for the preview image used for that touchpanel, taken from the corresponding <i>PreviewUrl.Unformatted[*]</i> signal. Tie this to the dynamic image object's URL source signal that shows the preview image.

Technical Tips and Tricks

1 Using the module – different device models, same module

There are (at the time of writing) 4 models of Epiphan Pearl: the Pearl Nano, Pearl Mini, Pearl-2, and Pearl Nexus. Each has a different amount of physical I/O and processing capability to support networked media and streaming media. However, the same Epiphan Pearl module has been designed to work with each model. This is for a few reasons:

- 1) Future compatibility – new product releases can be supported faster, and changes to existing products may be supported immediately depending on the nature of the change.
- 2) Ease of use within the program – no need to switch module versions for different models of Epiphan Pearl. Allows for greater flexibility when using the same program across multiple different AV systems that may have different needs.

As the Pearl product series has many features, this module is quite large. This section will address differences between the models and how they relate to the use of this module within your SIMPL Windows program.

Pearl-2

The Pearl-2 is the flagship product from Epiphan and supports all features offered by this module. It doesn't have a hard limit on the number of network inputs or the number of channels (Epiphan recommends a maximum of 6 channels). Any number of network inputs can be configured, provided there is enough processing power available.

The number of physical inputs can vary as well depending on if the 4K license feature is used.

The Pearl-2 will always have two outputs. It will have at least 10 inputs if the 4K license feature is active:

- 1) HDMI-A
- 2) HDMI-B
- 3) HDMI-4K-A
- 4) HDMI-4K-B
- 5) SDI-4K-A
- 6) SDI-4K-B
- 7) Analog-A Audio
- 8) Analog-B Audio
- 9) USB-A
- 10) USB-B

The Pearl-2 supports all 5 of the CMS providers supported by the module.

Pearl Mini

The Mini is a middle ground between the flagship Pearl-2 and Pearl Nano. Like the Pearl-2, there is no hard limit to the number of channels, but a maximum of 3 channels is recommended by Epiphan. It has much less physical inputs than the Pearl-2 as it does not support 4K. The number of network inputs, like the Pearl-

2, is not hard-limited by the device web UI. Any number can be configured without issue provided there is enough processing power available.

The Pearl Mini will always have one output. It will have at least 6 inputs:

- 1) HDMI-A
- 2) HDMI-B
- 3) SDI
- 4) Analog Audio
- 5) USB-A
- 6) USB-B

The Pearl Mini supports all 5 of the CMS providers supported by the module.

Pearl Nexus

The Nexus is extremely similar to the Epiphan Mini, except for it's physical 1RU formfactor. From a programming perspective, it can be treated mostly the same way. Consult the Pearl Mini section above for the number of channels, inputs, outputs, and other things to expect.

Pearl Nano

The largest difference is found with the Pearl Nano. As it only supports a single channel, the Pearl Nano does not support multitrack recorders, and the only recorder available is the one tied to its single channel.

The Pearl Nano will always have one channel, one recorder (linked to the channel) and one output. It will have at least 4 inputs:

- 1) HDMI
- 2) SDI
- 3) XLR/RCA Audio-Only input
- 4) USB Audio-Only

In addition to these 4 video and audio inputs, a limit of one network input can be configured. Web Graphics inputs are not supported by the Nano.

The Pearl Nano supports all 5 of the CMS providers supported by the module.

2 Preview images

To work around limitations the Crestron Construct image object has and bugs at the time of writing around the touchpanel prompting for image credentials when running a Crestron Construct CH5 UI; the Epiphan Pearl module will fetch the preview image every 5 seconds for the inputs, outputs, channels, and channel layouts when enabled and host them internally on the processors own web server.

This bypasses the need for image authentication and having to accept the self-signed certificate the images are hosted with when using HTTPS connections. This also allows for user interfaces to access preview images if the user interface is on a different network segment than the Epiphan Pearl in the case that Isolation Mode is required for the processor.

The output of the Epiphan Pearl module for the preview image will look something like this:

`/cws/epiphan/S-2.1.1:S-1/channel/1/preview.jpeg`

The full URL in your browser to download the image or for an image object on your UI will be something like this:

<http://10.0.10.55/cws/epiphan/S-2.1.1:S-1/channel/1/preview.jpeg>

As the Crestron processor may have multiple network interfaces, the touchpanel may have a different network address it accesses the processor on as part of the full URL when fetching the preview image. The touchpanel could be connected to the control subnet, on a network the Crestron processor has its LAN NIC connected with, etc. The “Epiphan Pearl Touchpanel Preview Url Helper” module described in the Complementary Modules section above can help format the output from the module as a full URL.

Additionally, the following steps are necessary to ensure the preview images work on a Crestron touchpanel running a CH5 project:

Crestron 4-series processors:

1. Users must run the *userpageauth off* command in an ssh console on the processor to allow the preview images to be served to the touchpanel without authentication
 - If authentication is required for the image, it will not work for Crestron Construct image objects, nor will it work for the Crestron Construct video object in mjpeg mode
 - If authentication is required for the image on a panel running a UI made in VTPro-e, it will work, but it will prompt for the credentials for each new image loaded. This is not a desirable experience.

Crestron 3-Series processors:

1. Ensure the processor is on a firmware version newer than or equal to 1.8001
 - Firmware versions older than 1.8001 have been observed to have problems with the CWS service.
 - As the image is served using the Crestron Web Services (CWS) running on the Crestron processor, it is strongly recommended to use a firmware version past 1.8001.
2. Run the *userpageauth off* command as you would for a 4-series processor.

3. Run the *ssl off* command.

- On a Crestron 3-series product, only processors without SSL enabled will serve the image without performing a 302 redirect to an https URL. Any 3-series processor past firmware version approx. 1.603 will have SSL enabled by default.
- This step is not required for 4-series processors as the web server on a 4-series processor will not force a redirect to use HTTPS when specifying HTTP in the URL.
- Please also note that when ssl is disabled on a 3-series processor, web XPanels will not work for the processor, as well as the setup web UI hosted on the processor. Therefore, it is recommended that SSL only be disabled if preview images are required *and* web xpanels are not required. Should these both be a requirement, a 4-series Crestron processor is recommended.
- This step is not strictly required if your user interface is running a VTPro-e project, but still recommended. Keep in mind that users will be prompted to accept the self-signed certificate for each preview image shown on the panel, and they will be required to do this on a regular basis. This not great UX and therefore turning SSL is the recommendation regardless.

4. Run the *reboot* command to apply the SSL change

- Warning: This will cause the processor to reboot, interrupting the running program.

Troubleshooting

1 A secure connection to the device is required

An IT group or 3rd party may require all devices on the network for the AV system use encrypted communications. The Epiphan Pearl does support HTTPS connections, by default on port 443, although it can be changed to a different port if desired.

Set the **UseHttps** parameter on the module to “Yes” or use the *UseHttps* Input signal to indicate HTTPS connections are to be used by the module. Then ensure the Pearl has been configured to use Https. Navigate to the Security page on the web UI and ensure the “Secure connection” checkbox is checked:

The screenshot shows the 'Security' configuration page in the Epiphan Pearl web UI. On the left is a sidebar menu with categories: Analog-A Audio, Analog-B Audio, USB-A, USB-B, Add input, Output ports (HDMI 1, HDMI 2), Events, Configuration (Automatic File Upload, Epiphan Edge, CMS, External USB drive, Network, Date and Time, Security, Touch Screen, External keyboard, Serial Port, USB Lights, Media, Maintenance), and Security (Touch Screen, External keyboard, Serial Port, USB Lights, Media, Maintenance). The 'Security' section is highlighted. The main content area is titled 'HTTPS configuration' and includes a 'Secure connection (HTTPS)' checkbox which is checked. Below this are fields for 'HTTP port' (80) and 'HTTPS port' (443), with a note that HTTP redirects to HTTPS when enabled. There are also fields for 'HTTP Keep-Alive timeout' (6 seconds) and 'HTTP Maximum Keep-Alive requests' (50). A 'Current Certificate' section shows 'Built-in self-signed certificate' and a 'Restore default certificate' button. A 'Certificate notice' box displays: 'Certificate notice: Issuer: O=Epiphan Video, Subject: O=Epiphan Video, From: 04/20/2026 To: 04/20/2046, Valid'. Below this are fields for 'New certificate' and 'Private Key', both with 'Choose file' buttons and 'No file chosen' text. An 'Apply' button is at the bottom. The 'CA certificates' section shows 'Built-in CA certificates' and 'Total certificates: 145' with a green checkmark.

When HTTPS is enabled on the Pearl, the module must be set to use https when connecting. If the HTTPS port used has been changed on this web page, ensure the Port being used by the module to connect on is changed accordingly as well.

2 Cannot connect to the device

Depending on how the system has been commissioned, there are a few things to check:

1) Is the username and password correct?

Access the Pearl web UI by entering its IP address (or hostname if the network has been setup with an appropriate DNS configuration) in a web browser. The browser will prompt for a username and password –

this can allow you to confirm if the username and password are valid. This same username and password can be used with the Epiphan Pearl module.

An admin user is recommended to be input to the module, as all functions are guaranteed to be available.

2) Is the port and connection mode being used by the module correct?

The module uses HTTP REST requests to communicate with the device. This is, by default, using HTTP port 80, but it is configurable. If it has been changed, the port value used by the module must match the "HTTP Port" or "HTTPS Port" field on the web UI and the "use https" value used by the module must match the state of the "Secure connection (HTTPS)" checkbox, seen on the "Security" page shown in the section above.

3) Is the Pearl able to communicate with the module?

The *Connection.Connected* signal indicates that regular communication is occurring with the Pearl and the username and password are correct. If you are sure the username, password, and port are set correctly, it is necessary to review if communication is occurring at all between the device and the module.

Enable debugging on the module at a "Verbose" level and view the output using Toolbox or an SSH client. Reviewing the output of the debug messages, especially if the logs are set to "verbose" will indicate quickly if communication is occurring properly between the Epiphan Pearl and the module. Regular operation will see REST requests exchanged between the module and the camera every few seconds.

If many "Error" messages are seen indicating a connection refusal or a timeout, this may indicate an issue with the AV network configuration. Troubleshoot this accordingly.

Once network configuration has been confirmed to not be an issue, if there is still an issue with device communications, please open a Support ticket with Epiphan, and include the debug logs captured, if available.