

This document provides additional assistance with wiring your Extron IP Link Pro Control Processor to your device. Different components may require a different wiring scheme than those listed below.

For complete operating instructions, refer to the user's manual for the specific IP Link Pro Control Processor or the documentation supplied by the manufacturer of the controlled device.

For more information on using Global Scriptor Modules, refer to the "[Guide to Using Scriptor Modules](#)" document.

Device Specifications

Device Type: Matrix Switcher
Manufacturer: WyreStorm
Firmware Version: MX-1007-HYB: 1.1.8
MX-0804-EDC: 1.1.0
Model(s): MX-1007-HYB, MX-0804-EDC

Tested on the Following Software and Firmware Versions

IP Link Pro Control Processor Firmware	ControlScript Deployment Utility Version
3.17.0000-b002	1.5.0-b.21

Version History

Module Version	Date	Notes
1_0_0	05/17/2024	Initial Version

Module Notes

- The following optional configuration variables are available. Any configuration variables being used should be set **prior** to calling the “Reinitialize” method on the module.
 - enableDebug: set to True to enable internal trace statements to be printed to the console. This is useful for debugging but should be set to False unless actively debugging.

Example

```
InterfaceName.enableDebug = True
```

- Call the “Reinitialize” method when ready to connect to the device. Once connected, the module will automatically handle reconnection (should connection be lost) and continue to try reconnecting until a connection is re-established.
- The module will automatically send periodic heartbeats (every 30 seconds) to the device to keep the connection alive. The module will consider itself disconnected if it does not receive 3 responses in a row. At that time, the module will automatically attempt to reconnect continuously every 5 seconds until a connection is re-established.

Supported Classes and Examples

SerialClass
InterfaceName = ModuleName.SerialClass(ProcessorName, 'COM1', Model='MX-0804-EDC')
EthernetClass
InterfaceName = ModuleName.EthernetClass('192.168.1.85', Model='MX-1007-HYB')

Set Commands

See demo program for usage examples.

Format with Qualifier:

```
InterfaceName.Set(Command, Value, {'Qualifier Key': 'Qualifier Value'})
```

Format without Qualifier:

```
InterfaceName.Set(Command, Value)
```

Command VideoRoute	Valid Values MX-1007-HYB: 'In 1 - 10' 'None' ¹ MX-0804-EDC: 'In 1 - 8' 'None' ¹
Qualifier Key Output	Valid Values MX-1007-HYB: 'Out 1 - 7' 'All' MX-0804-EDC: 'Out 1 - 4' 'All'
# Example Command Module.Set('Video Route', 'In 3', {'Output': 'Out 5'})	

¹ Sending 'None' will power down the output.

Command AudioMixerInputSource	Valid Values 'On' 'Off' 'Toggle'
Qualifier Key Mix	Valid Values 'Mix 1 - 2' 'USB Mixer'
Qualifier Key Source	Valid Values MX-1007-HYB: 'Out 1 - 7' 'Dante In 1 - 4' 'USB Host' ¹ 'Mic In 1 - 2' 'Line In' MX-0804-EDC: 'Out 1 - 4' 'Dante In 1 - 4' 'USB Host' ¹ 'Mic In 1 - 2' 'Line In'
# Example Command Module.Set('AudioMixerInputSource', 'On', {'Mix': 'Mix 1', 'Source': 'Out 1'})	

¹ 'USB Host' does not work if the mix is set to 'USB Mixer'

Command AudioMixerInputMute	Valid Values 'On' 'Off' 'Toggle'
Qualifier Key Mix	Valid Values 'Mix 1 - 2' 'USB Mixer'
Qualifier Key Source	Valid Values MX-1007-HYB: 'Out 1 - 7' 'Dante In 1 - 4' 'USB Host' ¹ 'Mic In 1 - 2' 'Line In' MX-0804-EDC: 'Out 1 - 4' 'Dante In 1 - 4' 'USB Host' ¹ 'Mic In 1 - 2' 'Line In'
# Example Command Module.Set('AudioMixerInputMute', 'Off', {'Mix': 'Mix 1', 'Source': 'Out 1'})	

¹ 'USB Host' does not work if the mix is set to 'USB Mixer'

Command AudioMixerInputVolume	Valid Values -20 to 20 -20 – 40 ¹
Qualifier Key Mix	Valid Values 'Mix 1 - 2' 'USB Mixer'
Qualifier Key Source	Valid Values MX-1007-HYB: 'Out 1 - 7' 'Dante In 1 - 4' 'USB Host' ² 'Mic In 1 - 2' 'Line In' MX-0804-EDC: 'Out 1 - 4' 'Dante In 1 - 4' 'USB Host' ² 'Mic In 1 - 2' 'Line In'
# Example Command Module.Set('AudioMixerInputVolume', -10, {'Mix': 'Mix 1', 'Source': 'Out 1'})	

¹ 'Mic In 1' and 'Mic In 2' sources have a range of -20 to 20. All other sources have a range of -20 to 40.

² 'USB Host' does not work if the mix is set to 'USB Mixer'

Command AudioMixerOutputSource	Valid Values 'On' 'Off' 'Toggle'
Qualifier Key Mix	Valid Values 'Mix 1 - 2'
Qualifier Key Source	Valid Values MX-1007-HYB and MX-0804-EDC: - Mix 1: 'Line Out 1' 'Amp Out' 'Dante Out 1/2' 'NHD Out' - Mix 2: 'Line Out 2' 'Dante Out 1/2'
# Example Command Module.Set('AudioMixerOutputSource', 'Off', {'Mix': 'Mix 2', 'Source': 'Line Out 1'})	

Command AudioMixerOutputMute	Valid Values 'On' 'Off' 'Toggle'
Qualifier Key Mix	Valid Values 'Mix 1 - 2' 'USB Mixer'
# Example Command Module.Set('AudioMixerOutputMute', 'Toggle', {'Mix': 'Mix 2'})	

Command AudioMixerOutputVolume	Valid Values -100 to 0
Qualifier Key Mix	Valid Values 'Mix 1 - 2' 'USB Mixer'
# Example Command Module.Set('AudioMixerOutputVolume', -25, {'Mix': 'Mix 2'})	

Command DisplayPower	Valid Values 'On' 'Off' 'Toggle'
Qualifier Key Display	Valid Values 'Out 1 - 6' 'All'
# Example Command Module.Set('DisplayPower', 'On', {'Display': 'Out 6'})	

Command PhantomPower	Valid Values 'On' 'Off' 'Toggle'
Qualifier Key Mic	Valid Values 1 - 2
# Example Command Module.Set('PhantomPower', 'On', {'Mic': 1})	

Command USBHost	Valid Values Type C USB Host 1 USB Host 2 In 9 In 10
# Example Command Module.Set('USBHost', 'Type C')	

Command RelayControl	Valid Values 'On' 'Off' 'Toggle'
Qualifier Key Relay	Valid Values 1 - 2
# Example Command Module.Set('RelayControl', 'Off', {'Relay': 1})	

Command SceneControl	Valid Values 'Save' 'Recall'
Qualifier Key Scene	Valid Values 1 - 6
# Example Command Module.Set('SceneControl', 'Recall', {'Scene': 3})	

Status Available

For all commands, call `SubscribeStatus` once in your program to register to receive feedback. Following this, call `Update` to query the device for its latest status. You will receive the status on the callback that has been assigned during the `SubscribeStatus` registration. Call `ReadStatus` to retrieve the last known status without re-querying the device.

See demo program for usage examples.

Format with Qualifier:

```
InterfaceName.Update(Command, {'Qualifier Key': 'Qualifier Value'})
Value = InterfaceName.ReadStatus(Command, {'Qualifier Key': 'Qualifier Value'})
InterfaceName.SubscribeStatus(Command, {'Qualifier Key': 'Qualifier Value'}, FeedbackHandler)
```

Format without Qualifier:

```
InterfaceName.Update(Command)
Value = InterfaceName.ReadStatus(Command)
InterfaceName.SubscribeStatus(Command, None, FeedbackHandler)
```

Command	Valid Values
CommunicationStatus	'Communicating' 'Not Communicating'
# Examples Module.SubscribeStatus('CommunicationStatus', None, CallbackFunction) Module.ReadStatus('CommunicationStatus')	

Command	Valid Values
InitializationStatus	'Initialized' 'Not Initialized'
# Examples Module.SubscribeStatus('InitializationStatus', None, CallbackFunction) Module.ReadStatus('InitializationStatus')	

Command	Valid Values
VideoRoute	MX-1007-HYB: 'In 1 - 10' 'None' MX-0804-EDC: 'In 1 - 8' 'None'
Qualifier Key	Valid Values
Output	MX-1007-HYB: 'Out 1 - 7' 'All' MX-0804-EDC: 'Out 1 - 4' 'All'
# Examples Module.SubscribeStatus('VideoRoute', None, CallbackFunction) Module.ReadStatus('VideoRoute', {'Output': 'Out 1'}) Module.Update('VideoRoute', {'Output': 'All'})	

Command AudioMixerInputSource	Valid Values 'On' 'Off'
Qualifier Key Mix	Valid Values 'Mix 1 - 2' 'USB Mixer'
Qualifier Key Source	Valid Values MX-1007-HYB: 'Out 1 - 7' 'Dante In 1 - 4' 'USB Host' ¹ 'Mic In 1 - 2' 'Line In' MX-0804-EDC: 'Out 1 - 4' 'Dante In 1 - 4' 'USB Host' ¹ 'Mic In 1 - 2' 'Line In'
# Examples Module.SubscribeStatus('AudioMixerInputSource', None, CallbackFunction) Module.ReadStatus('AudioMixerInputSource', {'Mix': 'Mix 1', 'Source': 'Mic In 1'}) Module.Update('AudioMixerInputSource', {'Mix': 'Mix 1'})	

¹ 'USB Host' does not work if the mix is set to 'USB Mixer'

Command AudioMixerInputMute	Valid Values 'On' 'Off'
Qualifier Key Mix	Valid Values 'Mix 1 - 2' 'USB Mixer'
Qualifier Key Source	Valid Values MX-1007-HYB: 'Out 1 - 7' 'Dante In 1 - 4' 'USB Host' ¹ 'Mic In 1 - 2' 'Line In' MX-0804-EDC: 'Out 1 - 4' 'Dante In 1 - 4' 'USB Host' ¹ 'Mic In 1 - 2' 'Line In'
# Examples Module.SubscribeStatus('AudioMixerInputMute', None, CallbackFunction) Module.ReadStatus('AudioMixerInputMute', {'Mix': 'Mix 1', 'Source': 'Mic In 1'}) Module.Update('AudioMixerInputMute', {'Mix': 'Mix 1', 'Source': 'Mic In 1'})	

¹ 'USB Host' does not work if the mix is set to 'USB Mixer'

Command AudioMixerInputVolume	Valid Values -20 to 20 ¹ -20 – 40 ²
Qualifier Key Mix	Valid Values 'Mix 1 - 2' 'USB Mixer'
Qualifier Key Source	Valid Values MX-1007-HYB: 'Out 1 - 7' 'Dante In 1 - 4' 'USB Host' ¹ 'Mic In 1 - 2' 'Line In' MX-0804-EDC: 'Out 1 - 4' 'Dante In 1 - 4' 'USB Host' ¹ 'Mic In 1 - 2' 'Line In'
# Examples Module.SubscribeStatus('AudioMixerInputVolume', None, CallbackFunction) Module.ReadStatus('AudioMixerInputVolume', {'Mix': 'Mix 1', 'Source': 'Mic In 1'}) Module.Update('AudioMixerInputVolume', {'Mix': 'Mix 1', 'Source': 'Mic In 1'})	

¹ 'Mic In 1' and 'Mic In 2' sources have a range of -20 to 20. All other sources have a range of -20 to 40.

Command AudioMixerOutputSource	Valid Values 'On' 'Off'
Qualifier Key Mix	Valid Values 'Mix 1 - 2'
# Examples Module.SubscribeStatus('AudioMixerOutputSource', None, CallbackFunction) Module.ReadStatus('AudioMixerOutputSource', {'Mix': 'Mix 2'}) Module.Update('AudioMixerOutputSource', {'Mix': 'Mix 2'})	

Command AudioMixerOutputMute	Valid Values 'On' 'Off'
Qualifier Key Mix	Valid Values 'Mix 1 - 2' 'USB Mixer'
# Examples Module.SubscribeStatus('AudioMixerOutputMute', None, CallbackFunction) Module.ReadStatus('AudioMixerOutputMute', {'Mix': 'Mix 2'}) Module.Update('AudioMixerOutputMute', {'Mix': 'Mix 2'})	

Command AudioMixerOutputVolume	Valid Values -100 to 0
Qualifier Key Mix	Valid Values 'Mix 1 - 2' 'USB Mixer'
# Examples Module.SubscribeStatus('AudioMixerOutputVolume', None, CallbackFunction) Module.ReadStatus('AudioMixerOutputVolume', {'Mix': 'Mix 1'}) Module.Update('AudioMixerOutputVolume', {'Mix': 'Mix 1'})	

Command PhantomPower	Valid Values 'On' 'Off'
Qualifier Key Mic	Valid Values 1 - 2
# Examples Module.SubscribeStatus('PhantomPower', None, CallbackFunction) Module.ReadStatus('PhantomPower', {'Mic': 1}) Module.Update('PhantomPower', {'Mic': 1})	

Command	Valid Values
USBHost	Type C USB Host 1 USB Host 2 In 9 In 10
# Examples Module.SubscribeStatus('USBHost', None, CallbackFunction) Module.ReadStatus('USBHost') Module.Update('USBHost')	

Command	Valid Values
RelayControl	'On' 'Off'
Qualifier Key	Valid Values
Relay	1 - 2
# Examples Module.SubscribeStatus('RelayControl', None, CallbackFunction) Module.ReadStatus('RelayControl', {'Relay': 2}) Module.Update('RelayControl', {'Relay': 2})	

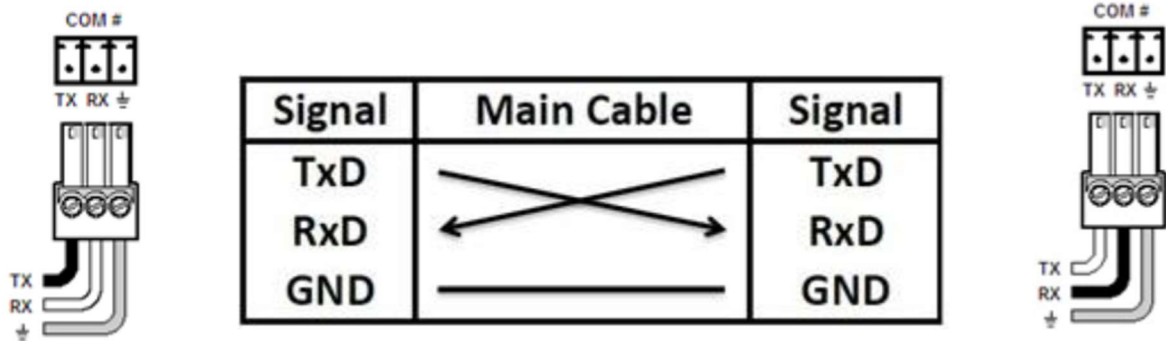
Serial Communication

Port Type: RS-232
Baud Rate: 9600
Data Bits: 8
Parity: None
Stop Bits: 1
Flow Control: Off

Cable and Adapter Requirements

Captive Screw to Captive Screw

Pin Assignments Diagram



Network Communication

Port Type:	Ethernet (TCP)
Default Port:	23
Logon Credentials Supported:	No
Multi-Connection Capabilities:	No
Port Changeability:	No

Module Support

Control Concepts, Inc.
(201) 797-7900
support@controlconcepts.net