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 Scripter Modules, refer to the "Guide to Using Scripter Modules" document.

Device Specifications

Device Type:	Display
Manufacturer:	Sharp
Firmware Version:	N/A
Model(s):	LC-70LE640U, LC-60LE640U, LC-70C6400U, LC-60C6400U, LC-52LE640U, LC-52C6400U, LC-80LE633U

Tested on the Following Software and Firmware Versions

IP Link Pro Control Processor Firmware	Global Scripter Version
2.06.0000-b010	1.4.1

Version History

Module Version	Date	Notes
1_0_1_1	9/7/2017	Updated status example.
1_0_1_0	8/21/2017	Initial Version

Module Notes

- Unidirectional variable must be set to 'True' if status is not required. Default value is 'False'.
  Example: `InterfaceName.Unidirectional = 'True'`

- connectionCounter variable must be set to the number of queries that will be sent to the device before displaying 'Disconnected' if no response is received. Default value is 15.
  Example: `InterfaceName.connectionCounter = 5`

- To Prevent intermittent lock up over time on the display, **ADD 10 seconds** delay between each Update() method call. See GS Example below for an example.

Supported Classes and Examples

SerialClass

```
InterfaceName = ModuleName.SerialClass(ProcessorName, 'COM1', Model='LC-70LE640U')
```

SerialOverEthernetClass

```
InterfaceName = ModuleName.SerialOverEthernetClass('192.168.254.254', 2001, Model='LC-70LE640U')
```

EthernetClass

```
InterfaceName = ModuleName.EthernetClass('192.168.254.254', 10002, Model='LC-70LE640U')
```

GS Example

Example below shows one way of implementing a polling loop with 10 seconds delay between each update query.

```
from extronlib.system import Wait
from extronlib.device import ProcessorDevice

# Import python built-in function cycle from itertools library
from itertools import cycle
import shrp_display_LC_xxC6400U_xxLE640U_xxLE633U_v1_0_1_0 as Sharp

Processor = ProcessorDevice('ProcessorAlias')
SharpDisplay = Sharp.SerialClass(Processor, 'COM1', Baud=9600, Model='LC-70LE640U')

# Creates an iterator with the list of Update commands that can be cycled through
query_list = cycle(['Power', 'AspectRatio', 'Input', 'Volume'])

# Function that calls Update() with the next element in the query_list
def send_query():

    # If the last element was called, next() will loop back and call the first element
    SharpDisplay.Update(next(query_list))
    polling_wait.Restart()

polling_wait = Wait(10, send_query)
```

Set Commands

Format with Qualifier:

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

Format without Qualifier:

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

Command	Value	Value	Value
AspectRatio	'Side Bar'	'S.Stretch'	'Zoom [AV]'
	'Stretch [AV]'	'Normal [PC]'	'Zoom [PC]'
	'Stretch [PC]'	'Dot by Dot [PC]'	'Full Screen [AV]'
	'Auto'	'Original'	

```
# AspectRatio example
InterfaceName.Set('AspectRatio', 'Side Bar')
```

Command	Value
AudioInput	None

```
# AudioInput example
InterfaceName.Set('AudioInput', None)
```

Command	Value	Value
AudioMute	'On'	'Off'

```
# AudioMute example
InterfaceName.Set('AudioMute', 'On')
```

Command	Value	Value
ChannelTV	'Up'	'Down'

```
# ChannelTV example
InterfaceName.Set('ChannelTV', 'Up')
```

Command	Value
ChannelTVCommand	'1' – '135'

```
# ChannelTVCommand example
InterfaceName.Set('ChannelTVCommand', 'String')
```

Command	Value
ClosedCaption	None

```
# ClosedCaption example
InterfaceName.Set('ClosedCaption', None)
```

Command	Value
DigitalAirCommand	'100' – '9999'

```
# DigitalAirCommand example
InterfaceName.Set('DigitalAirCommand', 'String')
```

Command	Value
DigitalCableCommand1	'0' – '9999'

```
# DigitalCableCommand1 example
```

```
InterfaceName.Set('DigitalCableCommand1', 'String')
```

Command	Value
DigitalCableCommand2	'0' – '6383'

```
# DigitalCableCommand2 example
InterfaceName.Set('DigitalCableCommand2', 'String')
```

Command	Value
DigitalCableMajorCommand	'1' – '999'

```
# DigitalCableMajorCommand example
InterfaceName.Set('DigitalCableMajorCommand', 'String')
```

Command	Value
DigitalCableMinorCommand	'0' – '999'

```
# DigitalCableMinorCommand example
InterfaceName.Set('DigitalCableMinorCommand', 'String')
```

Command	Value
Freeze	None

```
# Freeze example
InterfaceName.Set('Freeze', None)
```

Command	Value	Value	Value
Input	'HDMI 1'	'HDMI 2'	'HDMI 3'
	'HDMI 4'	'Component'	'Video 1'
	'Video 2'	'PC'	'TV'

```
# Input example
InterfaceName.Set('Input', 'HDMI 1')
```

Command	Value	Value	Value
MenuNavigation	'Up'	'Down'	'Left'
	'Right'	'Menu'	'Enter'
	'Exit'		

```
# MenuNavigation example
InterfaceName.Set('MenuNavigation', 'Up')
```

Command	Value	Value
Power	'On'	'Off'

```
# Power example
InterfaceName.Set('Power', 'On')
```

Command	Value
Volume	0 to 60 in steps of 1

```
# Volume example
InterfaceName.Set('Volume', 60)
```

Status Available

For all commands, call Update to receive the latest status. ConnectionStatus does not support the Update function and is triggered by the device providing a successful response to other Update function calls.

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)
```

FeedbackHandler will be called only when the specified qualifier gets a new status.

Format without Qualifier:

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

FeedbackHandler will be called when any qualifier gets a new status.

Command	Value	Value	Value
AspectRatio	'Side Bar'	'S.Stretch'	'Zoom [AV]'
	'Stretch [AV]'	'Normal [PC]'	'Zoom [PC]'
	'Stretch [PC]'	'Dot by Dot [PC]'	'Full Screen [AV]'
	'Auto'	'Original'	

```
# AspectRatio examples
InterfaceName.Update('AspectRatio')
Value = InterfaceName.ReadStatus('AspectRatio')
InterfaceName.SubscribeStatus('AspectRatio', None, FeedbackHandler)
```

Command	Value	Value
AudioMute	'On'	'Off'

```
# AudioMute examples
InterfaceName.Update('AudioMute')
Value = InterfaceName.ReadStatus('AudioMute')
InterfaceName.SubscribeStatus('AudioMute', None, FeedbackHandler)
```

Command	Value	Value
ConnectionStatus	'Connected'	'Disconnected'

```
# ConnectionStatus examples
Value = InterfaceName.ReadStatus('ConnectionStatus')
InterfaceName.SubscribeStatus('ConnectionStatus', None, FeedbackHandler)
```

Command	Value	Value	Value
Input	'HDMI 1'	'HDMI 2'	'HDMI 3'
	'HDMI 4'	'Component'	'Video 1'
	'Video 2'	'PC'	

```
# Input examples
InterfaceName.Update('Input')
Value = InterfaceName.ReadStatus('Input')
InterfaceName.SubscribeStatus('Input', None, FeedbackHandler)
```

Command	Value	Value
Power	'On'	'Off'

```
# Power examples
InterfaceName.Update('Power')
Value = InterfaceName.ReadStatus('Power')
InterfaceName.SubscribeStatus('Power', None, FeedbackHandler)
```

Command	Value
Volume	0 to 60 in steps of 1

```
# Volume examples
InterfaceName.Update('Volume')
Value = InterfaceName.ReadStatus('Volume')
InterfaceName.SubscribeStatus('Volume', None, FeedbackHandler)
```

Cable and Adapter Requirements

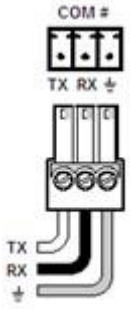
Captive Screw to Female DB9 RS-232 Serial Cable

Notes for the Device

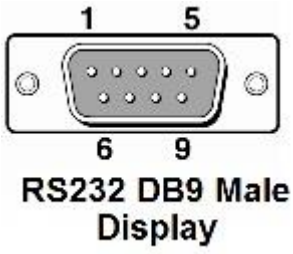
Serial communication

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

Pin Assignments Diagram



Signal	Main Cable	Pin	Signal
TxD	→	2	RxD
RxD	←	3	TxD
GND	—	5	GND



Network communication

When configuring the Ethernet module, be sure device settings match those of the Global Scripter ethernet interface

Port Type:	Ethernet
Default Port:	10002
Logon Credentials Supported:	No
Multi-Connection Capabilities:	No
Port Changeability:	Yes

Ethernet Module Configuration Description

Please refer to user manual for settings and changes to the network communication

Notes for the Device

Appendix A. Set Commands

Aspect Ratio Auto	WIDE10 \x0D
Aspect Ratio Dot by Dot [PC]	WIDE8 \x0D
Aspect Ratio Full Screen [AV]	WIDE9 \x0D
Aspect Ratio Normal [PC]	WIDE5 \x0D
Aspect Ratio Original	WIDE11 \x0D
Aspect Ratio S.Stretch	WIDE2 \x0D
Aspect Ratio Side Bar	WIDE1 \x0D
Aspect Ratio Stretch [AV]	WIDE4 \x0D
Aspect Ratio Stretch [PC]	WIDE7 \x0D
Aspect Ratio Zoom [AV]	WIDE3 \x0D
Aspect Ratio Zoom [PC]	WIDE6 \x0D
Audio Input None	ACHA0 \x0D
Audio Mute Off	MUTE2 \x0D
Audio Mute On	MUTE1 \x0D
ChannelTV Up	CHUP0 \x0D
ChannelTV Down	CHDW0 \x0D
ChannelTVCommand 1	DCCH001 \x0D
ChannelTVCommand 135	DCCH135 \x0D
Closed Caption None	CLCP0 \x0D
DigitalAirCommand 100	DA2P0100\x0D
DigitalAirCommand 9999	DA2P9999\x0D
DigitalCableCommand1 0	DC100000\x0D
DigitalCableCommand1 9999	DC109999\x0D
DigitalCableCommand2 0	DC110000\x0D
DigitalCableCommand2 6383	DC116383\x0D
DigitalCableMajorCommand 1	DC2U001 \x0D
DigitalCableMajorCommand 999	DC2U999 \x0D
DigitalCableMinorCommand 0	DC2L000 \x0D
DigitalCableMinorCommand 999	DC2L999 \x0D
Freeze None	RCKY54 \x0D
Input HDMI 1	IAVD1 \x0D
Input HDMI 2	IAVD2 \x0D
Input HDMI 3	IAVD3 \x0D
Input HDMI 4	IAVD4 \x0D
Input Component	IAVD5 \x0D
Input Video 1	IAVD6 \x0D
Input Video 2	IAVD7 \x0D
Input PC	IAVD8 \x0D
Input TV	IAVD0 \x0D
MenuNavigation Up	RCKY41 \x0D
MenuNavigation Down	RCKY42 \x0D
MenuNavigation Left	RCKY43 \x0D

MenuNavigation Right	RCKY44 \x0D
MenuNavigation Menu	RCKY38 \x0D
MenuNavigation Enter	RCKY40 \x0D
MenuNavigation Exit	RCKY46 \x0D
Power On	POWR1 \x0D
Power Off	POWR0 \x0D
Volume 0	VOLM000 \x0D
Volume 60	VOLM060 \x0D

Appendix B. Query Commands

Aspect Ratio	WIDE????\x0D
Audio Mute	MUTE????\x0D
Input	IAVD????\x0D
Power	POWR????\x0D
Volume	VOLM????\x0D