

## GENERAL INFORMATION

<b>CATEGORY</b>	Video switching
<b>VERSION</b>	1.0.0.4
<b>SUMMARY</b>	Controls the Intelix FLX Series via RS-232 or IP.
<b>GENERAL NOTES</b>	Controls routing of sources to destinations based on analog inputs. Provides polling routine with adjustable timing. Provides EDID management including reset and copying from outputs. Provides preset management including save, reset, and clear. Provides standby control. Provides route-through and deroute all functions. Provides front panel lock out. Provides asynchronous feedback message control. Provides per-input HDCP compliance control. Provides per-input input signal type control. Provides per-input VGA auto-adjust. Provides per-input scaling controls. Provides per-input HDMI audio embedding controls. Provides true feedback.
<b>COMMUNICATION SETUP</b>	RS232: 9600 Baud, No Parity, 8 Data Bits, 1 Stop Bit TCP/IP: Port 4001
<b>MODULE DEFINITION</b>	<pre>define_module 'Intelix FLX-Series v1_0_0_4' intelix(dvFLX88, vdvFLX88)</pre>

## CONTROL

<b>Poll All Outputs</b> Pulse to refresh routing feedback	Channel 1 pulse[vdv, 1]
<b>Route to Output n</b> Send level to route to Output	Level n send_level vdv, 1, 3
<b>Set Hostname to xxxxx</b> For IP control, set Hostname to connect to	Command 'HOST=xxxxx' send_command vdv, 'HOST=192.168.0.178'
<b>Set Port to nnnnn</b> For IP control, set Port to connect to	Command 'PORT=nnnnn' send_command vdv, 'PORT=4001'
<b>EDID Reset All</b> Reset all EDID signatures on Inputs to factory defaults	Command 'EDID RESET' send_command vdv, 'EDID RESET'
<b>Recall Preset n</b> Recall previously saved routing preset	Command 'PRESET RECALL=nn' send_command vdv, 'PRESET RECALL=7'

**Store Preset n**

Store routing preset for later recall

Command 'PRESET STORE=nn'

send\_command vdv, 'PRESET STORE=5'

**Clear Preset n**

Erase routing preset

Command 'PRESET CLEAR=nn'

send\_command vdv, 'PRESET CLEAR=3'

**Power On**

Power unit on

Pulse Channel 27

pulse[vdv, 27]

**Power Off**

Power unit off

Pulse Channel 28

pulse[vdv, 28]

**Route all Inputs Through**

Route input 1 to Output 1, 2 to 2, 3 to 3, etc.

Command 'ROUTES=ALL THROUGH'

send\_command vdv, 'ROUTES=ALL THROUGH'

**Unroute all Outputs**

Clear all routing assignments

Command 'ROUTES=ALL CLEAR'

send\_command vdv, 'ROUTES=ALL CLEAR'

**Front Panel Lock**

Disable commands entered on front panel

Pulse Channel 70

pulse[vdv, 70]

**Front Panel Unlock**

Enable commands entered on front panel

Pulse Channel 71

pulse[vdv, 71]

**Feedback Messages On**

Enable front panel routing feedback

Pulse Channel 72

pulse[vdv, 72]

**Feedback Messages Off**

Disable front panel routing feedback

Pulse Channel 73

pulse[vdv, 73]

**Copy EDID for Input from Output**

Assign EDID from Output oo to Input ii

Command 'INPUT ii EDID=oo'

send\_command vdv, 'INPUT 01 EDID=03'

**Enable HDCP Compliance for Input nn**

Allow input card to negotiate HDCP security

Command 'INPUT nn HDCP=ON'

send\_command vdv, 'INPUT 01 HDCP=ON'

**Disable HDCP Compliance for Input nn**

Prohibit input card from negotiating HDCP security

Command 'INPUT nn HDCP=OFF'

send\_command vdv, 'INPUT 01 HDCP=OFF'

**Set Input nn Signal Type to VGA**

Set Input Signal Type to VGA

Command 'INPUT nn RI4 SIGNAL=VGA'

send\_command vdv, 'INPUT 01 RI4  
SIGNAL=VGA'

**Set Input nn Signal Type to Component**  
Set Input Signal Type to Component (YPbPr)

```
Command 'INPUT nn RI4 SIGNAL=YPBPR'  
send_command vdv, 'INPUT 01 RI4  
SIGNAL=YPBPR'
```

**Set Input nn Signal Type to S-Video**  
Set Input Signal Type to S-Video

```
Command 'INPUT nn RI4 SIGNAL=S-VIDEO'  
send_command vdv, 'INPUT 01 RI4  
SIGNAL=S-VIDEO'
```

**Set Input nn Signal Type to Composite**  
Set Input Signal Type to Video

```
Command 'INPUT nn RI4 SIGNAL=VIDEO'  
send_command vdv, 'INPUT 01 RI4  
SIGNAL=VIDEO'
```

**Trigger VGA Auto-adjust on Input nn**  
Cause input card to analyze VGA signal for alignment

```
Command 'INPUT nn RI4 AUTOADJUST'  
send_command vdv, 'INPUT 01 RI4  
AUTOADJUST'
```

**Set Input nn Scaler Resolution to 1024x768**  
Configure internal scaler for XGA resolution

```
Command 'INPUT nn RI4 SCALER=XGA'  
send_command vdv, 'INPUT 01 RI4  
SCALER=XGA'
```

**Set Input nn Scaler Resolution to 1280x720**  
Configure internal scaler for 720p resolution

```
Command 'INPUT nn RI4 SCALER=720P'  
send_command vdv, 'INPUT 01 RI4  
SCALER=720P'
```

**Set Input nn Scaler Resolution to 1280x800**  
Configure internal scaler for WXGA resolution

```
Command 'INPUT nn RI4 SCALER=WXGA'  
send_command vdv, 'INPUT 01 RI4  
SCALER=WXGA'
```

**Set Input nn Scaler Resolution to 1920x1080**  
Configure internal scaler for 1080p resolution

```
Command 'INPUT nn RI4 SCALER=1080P'  
send_command vdv, 'INPUT 01 RI4  
SCALER=1080P'
```

**Set Input nn Audio Source to Digital**  
Configure HDMI input to capture HDMI audio

```
Command 'INPUT nn HI4A AUDIO=DIGITAL'  
send_command vdv, 'INPUT 01 HI4A  
AUDIO=DIGITAL'
```

**Set Input nn Audio Source to Analog**  
Configure HDMI input to capture external audio

```
Command 'INPUT nn HI4A AUDIO=ANALOG'  
send_command vdv, 'INPUT 01 HI4A  
AUDIO=ANALOG'
```

## FEEDBACK

**Input n Routed to Output x**

True feedback, reports when routing changes

String 'OUT xx: nn'

'OUT 01: 03'

**Unit Powered On**

True feedback, unit powered on

String 'POWER: ON'

'POWER: ON'

**Unit Powered Off**

True feedback, unit powered off

String 'POWER: OFF'

'POWER: OFF'

**Front Panel is Locked**

True feedback, front panel locked

String 'FRONT PANEL:LOCKED'

'FRONT PANEL: LOCKED'

**Front Panel is Unlocked**

True feedback, front panel unlocked

String 'FRONT PANEL:UNLOCKED'

'FRONT PANEL: UNLOCKED'

**Feedback Messages Are On**

True feedback, feedback messages are on

String 'FEEDBACK: ON'

'FEEDBACK: ON'

**Feedback Messages Are Off**

True feedback, feedback messages are off

String 'FEEDBACK: OFF'

'FEEDBACK: OFF'