Wheatstone

Interfacing House Routers Using Translator

Translator Software versions 4.5x

Version 2.0 / 10-4-2006

1.0 Introduction

The following document describes the installation and configuration of a software interface that integrates house router control with a Wheatstone Bridge or Wheatnet Digital Audio Network. Control of the house router can be initiated from a control surface or any other Wheatstone XY controller. The software also monitors the house router status and updates any appropriate Wheatstone displays.

2.0 Theory of Operation

Wheatstone control surfaces and XY controllers offer complete control of the Digital Audio Network. A natural extension of this control philosophy leads to interfacing 3rd party systems. Due to the natural disparities between various manufacturers designs and “languages”, a way to bridge the communication gap is required if 3rd party control is to be realized. Wheatstone’s Translator software application bridges that gap.

In a traditional analog system, it is common for an audio operator to pot up a REMOTE fader when doing a live shot. The audio actually feeding that fader is typically routed to it by a house router control head either in the audio booth or perhaps the control room. By integrating house router control directly from the audio control surface, a cleaner, intuitive user interface is established.

Enter Translator. Behind the scenes, a PC connected to an Ethernet LAN running a Wheatstone application called Translator, monitors all connection traffic in BOTH systems and sends a connection message to the house router when an operator "TAKES" a “house” source to a control surface input fader. Surface displays will also update if a change is made from the house router end.

Some configuration work is done up front in XPoint to allow Translator to make the correct source and destination route in the house router. Generally speaking, the system is configured as follows:

1- Edit the Translator ini file to specify a house router type, IPaddress , IP Port, and level mask.
2- Provide a fixed number of destinations (outputs) on the house router that physically connect to sources (inputs) on the Wheatstone router.
3- These house Destinations are mapped into new SOURCE signals using XPoint and given the attribute "Router Interconnect".
4- You provide a list of house source signal names and their signal ID #'s (include all audio sources you want board operators to see).
5- These house sources are entered into XPoint as new Source signals and given the attribute "3rd Party Router".

From an operator’s perspective, they simply scroll and take a house source, just like a Wheatstone source. Last minute script change? No problem. House sources become just another resource on any control surface fader. No special training is required.

Translator’s Role

Remember that Translator is running in the background, quietly listening to house router and Wheatstone router cross-point requests. When an operator takes a house source on a control surface, Translator knows, from the 3rd Party Router attribute flags, that the source requested is a house router source. Translator then assembles a Take message using the level and signal ID numbers you configured and sends it over Ethernet to the house router. The display on the control surface fader shows the audio signal alias name you entered in XPoint

What physical cross-point is made? You must decide up front how many physical house router audio destinations will be wired into the Wheatstone system. These may be analog or AES-3 digital audio. Typically the number falls into the 2 to 16 range but depends on the number of free outputs you have on the house router. These destinations are configured as *sources in the Wheatstone system* and are given the attribute *Router Interconnect.* This set of signals becomes a pool of destinations for house router source signals. When you TAKE a house router Source to a fader, it will actually be routed to one of these Router Interconnect inputs. Router Interconnects are invisible to the audio operator. What they “see” on their input fader display is the house router source signal name. Obviously the more

Router Interconnect signals you have available, the more simultaneous house router sources may be routed to a control surface.

3.0 Hardware Requirements

- Ethernet interface - Translator speaks and listens over Ethernet networks exclusively. There are no provisions for serial interfacing to house routers except the GV Horizon.
- PC running Translator – 100Mbps NIC, Windows XP Service Pack 2 or Windows 2000 Service Pack 4

3.1 Supported House Routers

The following routers are currently supported.

- Grass Valley SMS-7000 – GV Native Protocol
- Grass Valley Encore System Controller - GV Native or RCL protocols
- Utah 300 and 400 series
- Utah SC-4 System Controller
- nVision 5128

3.2 Multiple House Router Support

Translator supports simultaneous connections to up to four house routers. House routers may be from different manufacturers.

4.0 Translator Software Installation

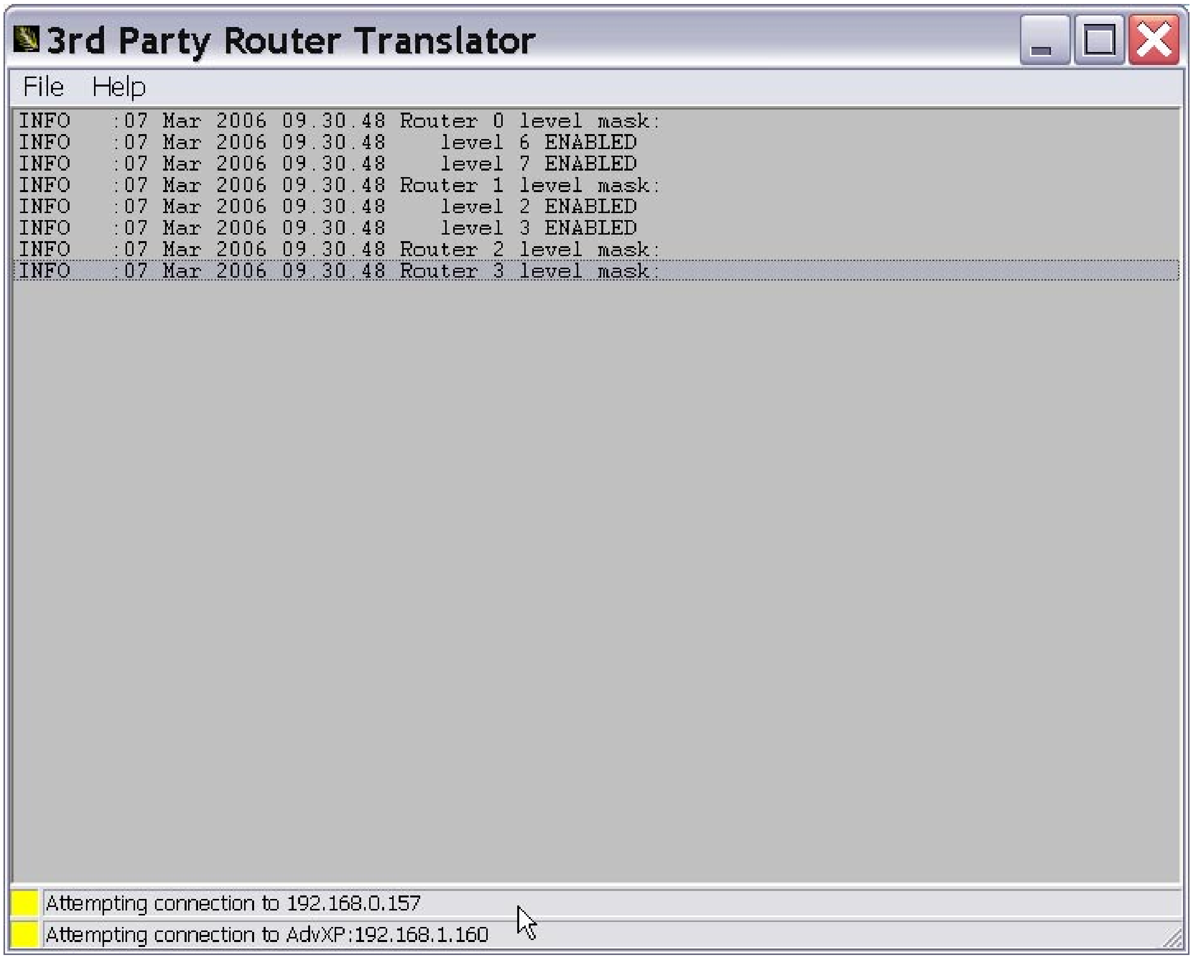
Translator is a mini application that runs in the background and acts like a driver. It uses a simple installer program and requires little more than text file editing to configure it.

- Double click on the Translator Installer file Translator.msi to install the application.
- Translator will be installed into the c:\Program Files\Wheatstone\Translator folder.
- The application is Translator.exe
- Once installed, a shortcut to Translator .exe should be included in the Windows Startup group so that the program starts whenever the PC is re-booted.
- Because Translator is a background application requiring no user intervention, Translator will start minimized in the System Tray.
- An ICON in the lower right corner of the Windows Task Bar provides access to the main Translator form for status and activity window.

4.1 What You Will See at Startup

When Translator first starts it will attempt to connect to the first house router at the IP address specified in Translator.ini. Translator will also attempt to connect to the Wheatstone Bridge or Wheatnet CPU at the IP address you have specified. The two yellow boxes will change to a solid green color when the required connections have been made. The main screen shows the communication activity and will echo all house router takes once communications have been established.

In this particular startup, as shown in Figure 1, the translator.ini file has two Utah router instances enabled. The first instance at Router ID 1 enables levels 6 and 7 and the second instance at Router ID 2 enables levels 2 and 3. Translator is attempting to connect to the Utah controller at 192.168.0.157 and the Wheatstone Bridge CPU (AdvXP) at 192.168.1.160.



```
INFO   :07 Mar 2006 09.30.48 Router 0 level mask:
INFO   :07 Mar 2006 09.30.48    level 6 ENABLED
INFO   :07 Mar 2006 09.30.48    level 7 ENABLED
INFO   :07 Mar 2006 09.30.48 Router 1 level mask:
INFO   :07 Mar 2006 09.30.48    level 2 ENABLED
INFO   :07 Mar 2006 09.30.48    level 3 ENABLED
INFO   :07 Mar 2006 09.30.48 Router 2 level mask:
INFO   :07 Mar 2006 09.30.48 Router 3 level mask:
```

🟨 Attempting connection to 192.168.0.157

🟨 Attempting connection to AdvXP:192.168.1.160

Figure 1 Translator's Main Screen

4.2 After Translator is Installed

Run the application once by double clicking on Translator.exe, then exit. This will create a default translator.ini file that you can then edit.

Note that you must edit the translator ini file before Translator can communicate with your house router.

4.3 Translator.INI Editing

You must edit the Translator.ini file so Translator knows which type of router it is talking and listening to. To edit the file be sure to Exit Translator BEFORE editing the file.
Important: Text editing is case sensitive. Don't add any extra characters and do not delete lines. You may add comments by inserting the "#" character at the start of the line. Make a backup copy in case you make a mistake.

The following instructions assume you only have one router interface, in this Example we'll setup a Grass Valley router with an Encore controller, but the steps are the same for any type of supported router.

4.4 Editing Example

1.0 Open the translator.ini file in Notepad or another text editor.
2.1 Edit the line: *NumRouters=1*
//this tells Translator how many house routers are being connected, just one in this case.

2.2 Edit the line: *Router1Interface=GrassValley*
//this sets the router interface 1 to be the Grass Valley Encore controller using RCL protocol.

2.3 Edit the line: *Router2Interface=* // clears the name and eliminates unwanted traffic.
2.4 Edit the line: *Router3Interface=* // clears the name and eliminates unwanted traffic.
2.5 Edit the line: *Router4Interface=* // clears the name and eliminates unwanted traffic.

3.0 Edit the line as required:
Router1LevelMask = (calculate level mask)
//where the 32 bit hexadecimal number entered equals the 32 bit level mask.

Ok, this is where you dust off your hexadecimal math skills. Translator uses a 32-bit level mask formatted as eight, 4-bit nibbles displayed in hexadecimal notation. Each nibble can be in the range form zero to F. Calculating the 32 bit hexadecimal level mask is a straight forward process; the first step is easy.

- First, identify which house router levels you want to switch audio on.
- Refer to the Level Mask Bit Weight chart to correlate the chosen levels to bit weights and mask nibble position.
- Add up the decimal bit weights separately *for each mask nibble*.
- Convert each decimal "sum of bit weights" to hexadecimal.
- Assemble each nibble into the "Router1LevelMask=" string. The hex string must start with a 0x followed by 8 hexadecimal characters (e.g. 0x87654321).

Level	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bit Weight	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8
Mask Nibble	1				2				3				4			
Level	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit Weight	1	2	4	8	1	2	4	8	1	2	4	8	1	2	4	8
Mask Nibble	5				6				7				8			

Table 1: Level Mask Bit Weight

Example1:

- For example, let's assume you want to switch audio on levels 2 and 3.
- The bit weights are level 2 =2 and level 3=4. Both are part of byte 1
- Add up the bit weights for Mask Byte 1: 2+4 = 6.
- Convert to hexadecimal, 6 = 0x6. The rest of the Mask Bytes are zeros.
- Assemble the 8 bytes with the 0x prefix: 0x00000006
- Edit Translator.ini file line: Router1LevelMask=0x00000006

Example 2:

- That was easy, how about levels 15 and16 and 32?
- The bit weights are level15 = 4 and level 16 = 8. Level 15 & 16 is byte 4
- The bit weight for level 32 is byte 8.
- Add up the bit weights for Mask Byte 4: 4+8=12.
- The bit weight for Mask Byte 8 is 8.
- Convert to hexadecimal, 12= 0xC and 8 = 0x8.
- Assemble the 8 bytes with the 0x prefix: 0x8000000C
- Edit Translator.ini file line: *Router1LevelMask*=0x8000000C

4.5 Finishing up the INI File

4.0 Edit the line under the [Grass Valley]: IPAddr=x.x.x.x //enter the Encore controller's IP address here
5.0 Edit the line under the [Grass Valley]: Port=12345 //12345 is the default GV port#, zero disables the interface.
5.1 Make sure all other unused router ports defined are set to zero. This will eliminate any wanted traffic.
6.0 Save the file!
6.1Restart Translator

Example Translator.ini file

```
[Routers]
#Valid Interfaces'Utah','GrassValley','GrassValleyNative' and 'nVision'
#Grassvalley = Encore controller RCL protocol , GrassValleyNative = Native protocol

NumRouters=3

Router1Interface=Utah
Router1LevelMask=0x0000000C

Router2Interface=Utah
Router2LevelMask=0x00000006

Router3Interface=GrassValley
Router3LevelMask=0x0000007

Router4Interface=
Router4LevelMask=0x00000000

[AdvXP]
#Wheatstone Bridge or Wheatnet
AdvXP_IPAddr=192.168.1.160

[Utah]
IPAddr=192.168.1.157
#Port=5001 default Utah port
# set Port to zero to disable
Port=5001
PollingRate=5

[GrassValley]
IPAddr=192.168.1.131
#Port=12345 default Encore RCL port
# set Port to zero to disable
Port=12345
PollingRate=1

[GrassValleyNative]
IPAddr=192.168.1.131
#Port=12345 default GV native port
# set Port to zero to disable
Port=0
PollingRate=1

[nVision]
IPAddr=192.168.1.179
#Port=9193 default nVision port
# set Port to zero to disable
Port=0
PollingRate=0.5
```

4.6 Translator.ini File Parameters

[Routers]

Parameter	Description
NumRouters	Valid values include 1-4. Designates how many routers are enabled. Correlates to the Router ID field when creating signals in XPoint.
RouterNInterface	Where N is the Router ID (1-4), valid values include the text strings ‘Utah’ or ‘GrassValley’, ‘GrassValleyNative’, and ‘nVision’. More options will become available when more interface types are supported. This describes the ‘type’ of interface to use to communicate with the router. Multiple routers can use the same interface.
RouterNLevelMask	Where N is the Router ID (1-4), valid values are any unsigned 32 bit integer. This field is a bit mask that enables certain levels for Router N. In the example above Router1LevelMask value 0x0000000C enables levels 3,4. This may be entered in Hexadecimal format but will be re-written as decimal on subsequent viewings of this ini file.

[Advxp]

Parameter	Description
AdvXP_IPAddr	Valid Internet Protocol address of the Wheatstone AdvXP Crosspoint Switch.

[Utah]

Parameter	Description
IPAddr	Valid Internet Protocol address of the Utah Scientific SC3 system controller.
Port	Valid values include 0, 5001. Use 0 to disable the interface and 5001 to open communication with the Utah Scientific SC3 system controller.

[GrassValley]

Parameter	Description
IPAddr	Valid Internet Protocol address of the Grass Valley Encore system controller.
Port	Valid values include 0, 12345. Use 0 to disable the interface and 12345 to open communication with the Grass Valley Encore system controller.

[GrassValleyNative]

Parameter	Description
IPAddr	Valid Internet Protocol address of the GrassValley RCL system controller.
Port	Valid values include 0, 12345. Use 0 to disable the interface and 12345 to open communication with the Grass Valley Encore system controller.

[nVision]

Parameter	Description
IPAddr	Valid Internet Protocol address of the nVision Housenet NIC.
Port	Valid values include 0, 9193. Use 0 to disable the interface and 9193 to open communication with the nVision Housenet NIC.

4.7 Hexadecimal Notation

Decimal	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Hex	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

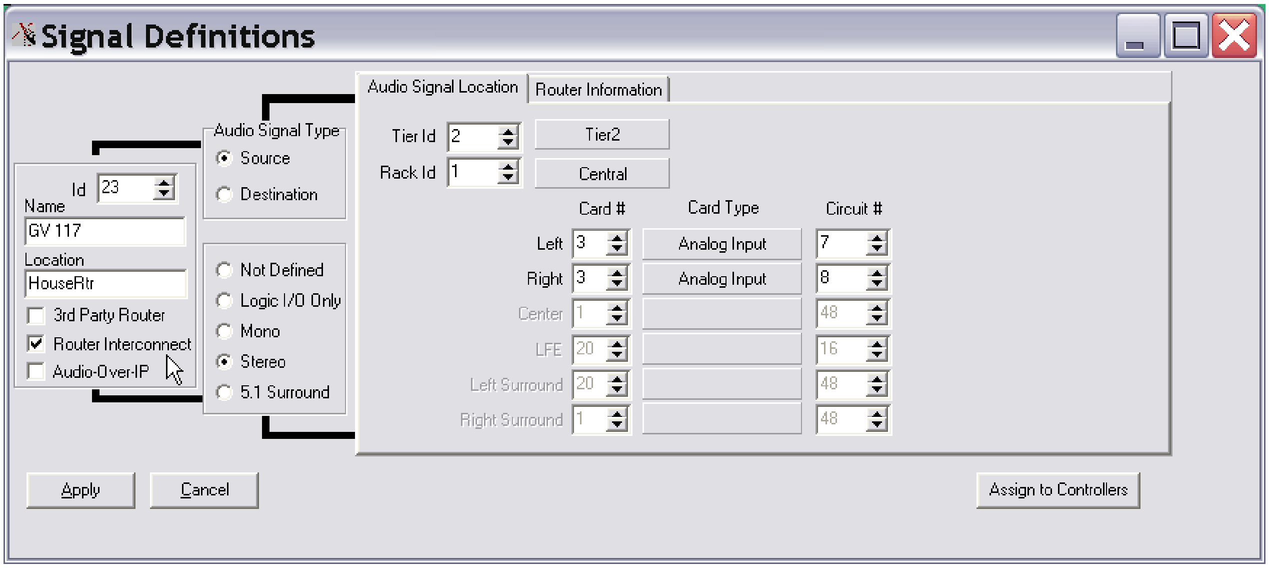
Table 2: Decimal to hexadecimal translation

5.0 Configuring House Router Signals in XPoint

There are three types of house router signals that may be created in XPoint. Each signal performs a different function. You use the Modify Signal Definitions form in XPoint to create each type of signal. To open the Signal Definitions form, right click on an existing signal or a blank Source signal ID location.

5.1 Router Interconnects

These signals are the actual physical audio connections between the house router destinations and the Wheatstone inputs. These signals may be analog or digital audio types.
To create a new *Router Interconnect* signal, right click on the existing audio *Source* signal that will physically connect to the house router destination. The following form appears:



Id 23
Name GV 117
Location HouseRtr
□ 3rd Party Router
☑ Router Interconnect
□ Audio-Over-IP

Audio Signal Type
● Source
○ Destination

○ Not Defined
○ Logic I/O Only
○ Mono
● Stereo
○ 5.1 Surround

Tier Id 2 Tier2
Rack Id 1 Central

	Card #	Card Type	Circuit #
Left	3	Analog Input	7
Right	3	Analog Input	8
Center	1		48
LFE	20		16
Left Surround	20		48
Right Surround	1		48

Figure 2 – Signal Definitions for Router Interconnect – Audio Setup

Remember, you are creating an audio input signal on the Wheatstone side that will connect to a house router destination. You need to do two things:

- Define the stereo source signal to be a *Router Interconnect.*
- Map the House Router ID, Router Destination number, and Level information to this audio input.

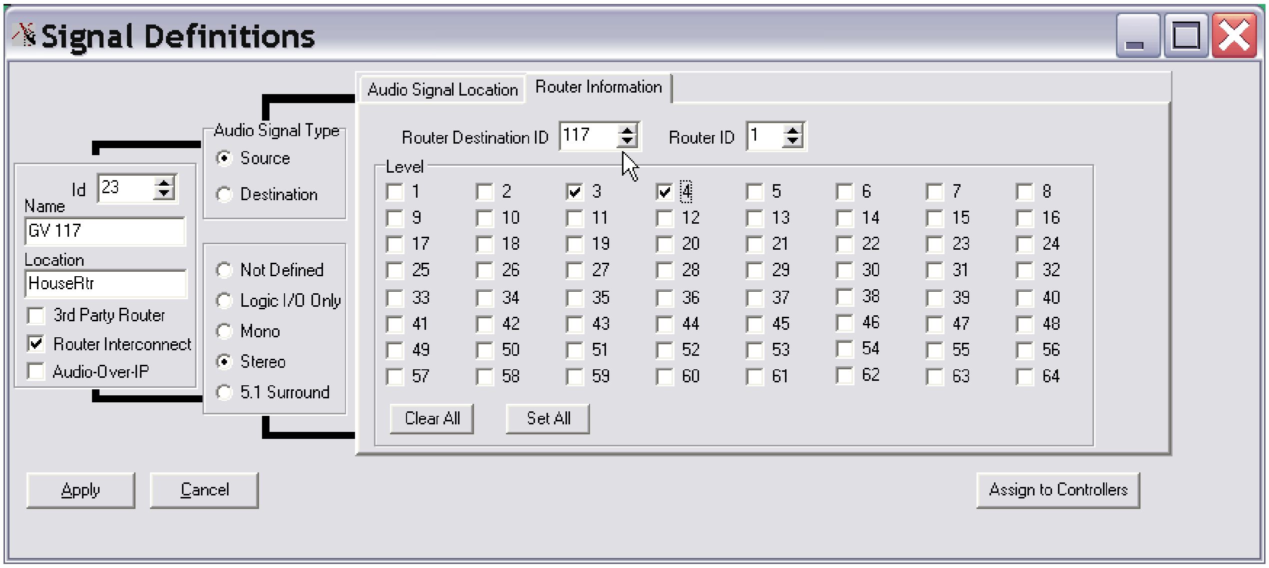
Audio Signal Location Tab

In this example, *GV 117* is the stereo, analog *Router Interconnect* signal connected to Tier 2, Rack1, Analog Input Card 3, Circuits 7&8. You will physically wire this Wheatstone input to the House Router destination you choose.

Router Information Tab – (see Figure 3 - next page)

In this example, GV 117 Router Destination ID is 117, and the Router ID is 1, levels 3 and 4 will be switch to destination 117.

Router Destination – this is the house router destination signal ID number.
Router ID – this number corresponds to the Router1Interface line in the Translator.ini file.
Level- Check the levels you want to switch with this signal.



Signal Definitions

Id 23

Name GV 117

Location HouseRtr

☐ 3rd Party Router

☑ Router Interconnect

☐ Audio-Over-IP

Audio Signal Type: ● Source ○ Destination

○ Not Defined ○ Logic I/O Only ○ Mono ● Stereo ○ 5.1 Surround

Audio Signal Location | Router Information

Router Destination ID 117 Router ID 1

Level

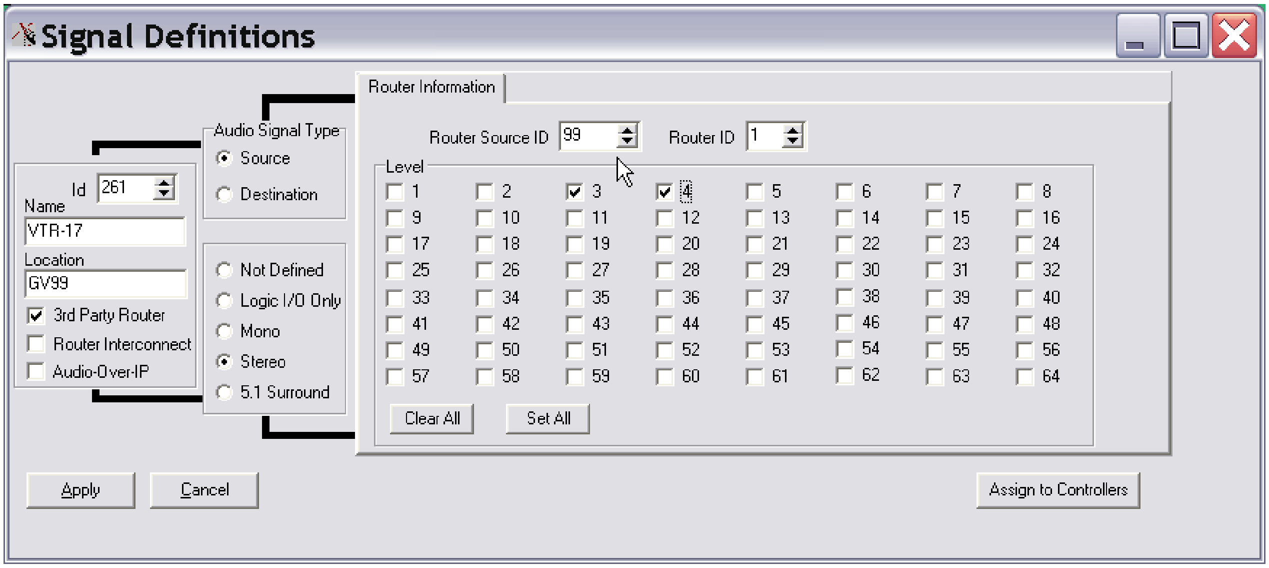
☐ 1	☐ 2	☑ 3	☑ 4	☐ 5	☐ 6	☐ 7	☐ 8
☐ 9	☐ 10	☐ 11	☐ 12	☐ 13	☐ 14	☐ 15	☐ 16
☐ 17	☐ 18	☐ 19	☐ 20	☐ 21	☐ 22	☐ 23	☐ 24
☐ 25	☐ 26	☐ 27	☐ 28	☐ 29	☐ 30	☐ 31	☐ 32
☐ 33	☐ 34	☐ 35	☐ 36	☐ 37	☐ 38	☐ 39	☐ 40
☐ 41	☐ 42	☐ 43	☐ 44	☐ 45	☐ 46	☐ 47	☐ 48
☐ 49	☐ 50	☐ 51	☐ 52	☐ 53	☐ 54	☐ 55	☐ 56
☐ 57	☐ 58	☐ 59	☐ 60	☐ 61	☐ 62	☐ 63	☐ 64

Figure 3 – Signal Definitions for Router Interconnect – Router Information Setup

5.2 House Sources

These Wheatstone signals are aliases to the Sources on the house router. Include all the Sources you want audio operators to have access to at the control surface. You will need to make new 8 character signal names and enter the house router signal ID number for each one. It is useful to print out a list of house router signal names and ID numbers before you begin programming.

To create a new *3rd Party Router* signal, right click on a *blank* audio *Source* signal location and select Modify Signal Definitions. The following form appears:



Signal Definitions

Id 261

Name VTR-17

Location GV99

☑ 3rd Party Router

☐ Router Interconnect

☐ Audio-Over-IP

Audio Signal Type: ● Source ○ Destination

○ Not Defined ○ Logic I/O Only ○ Mono ● Stereo ○ 5.1 Surround

Router Information

Router Source ID 99 Router ID 1

Level

☐ 1	☐ 2	☑ 3	☑ 4	☐ 5	☐ 6	☐ 7	☐ 8
☐ 9	☐ 10	☐ 11	☐ 12	☐ 13	☐ 14	☐ 15	☐ 16
☐ 17	☐ 18	☐ 19	☐ 20	☐ 21	☐ 22	☐ 23	☐ 24
☐ 25	☐ 26	☐ 27	☐ 28	☐ 29	☐ 30	☐ 31	☐ 32
☐ 33	☐ 34	☐ 35	☐ 36	☐ 37	☐ 38	☐ 39	☐ 40
☐ 41	☐ 42	☐ 43	☐ 44	☐ 45	☐ 46	☐ 47	☐ 48
☐ 49	☐ 50	☐ 51	☐ 52	☐ 53	☐ 54	☐ 55	☐ 56
☐ 57	☐ 58	☐ 59	☐ 60	☐ 61	☐ 62	☐ 63	☐ 64

Figure 4 – Signal Definitions for 3rd Party Router aliases – Router Information Setup

When you create 3rd Party Router Sources in XPoint you are creating aliases to the house source signals. Notice that there is no Audio information tab, only a Router Information tab. To configure, enter a house source name (it should be similar to the house name). Also check the 3rd Party Router checkbox. Now enter the house router signal ID# and Router ID.

Router Source ID – this number is the house router signal ID for this source.
Router ID - this number corresponds to the Router1Interface line in the Translator.ini file. Translator supports simultaneous control of up to 4 different house routers.

5.3 House Destinations

These signals are rarely used but Translator may be used to control cross-points entirely in the house router. In fact complete control of the house router source and destination map can be obtained by mapping all source and destinations in the house router as 3rd Party Router signals in XPoint. Then Wheatstone XY controllers can be used as control heads for the house router. These signals are aliases to the Destinations on house router. Include all the Destinations you want operators to have access to. You will need to make 8 character signal names and enter the house router signal ID number for each one. It is useful to print out a list of house router signal names and ID numbers before you begin programming.

6.0 Configuring House Routers for External Control

In most cases changes must be made to your house router configuration in order to enable external control from Wheatstone's Translator application. Please refer to your house router documentation and vendor for details on implementing external control over Ethernet.

Grass Valley SMS 7000

- Use the "GrassValleyNative" Translator router type.
- no special programming is required.
- Source and destinations are zero based even thought the terminal reports them as 1 based.

Grass Valley Encore Controller

- Use the "GrassValley" Translator router type.
- Requires a GV sharing software application be installed and running.
- Requires that a Control Panel Server of the type RCL be defined.
- Check GV documentation for server configuration details.
- Source and destinations are zero based even thought the terminal reports them as 1 based.

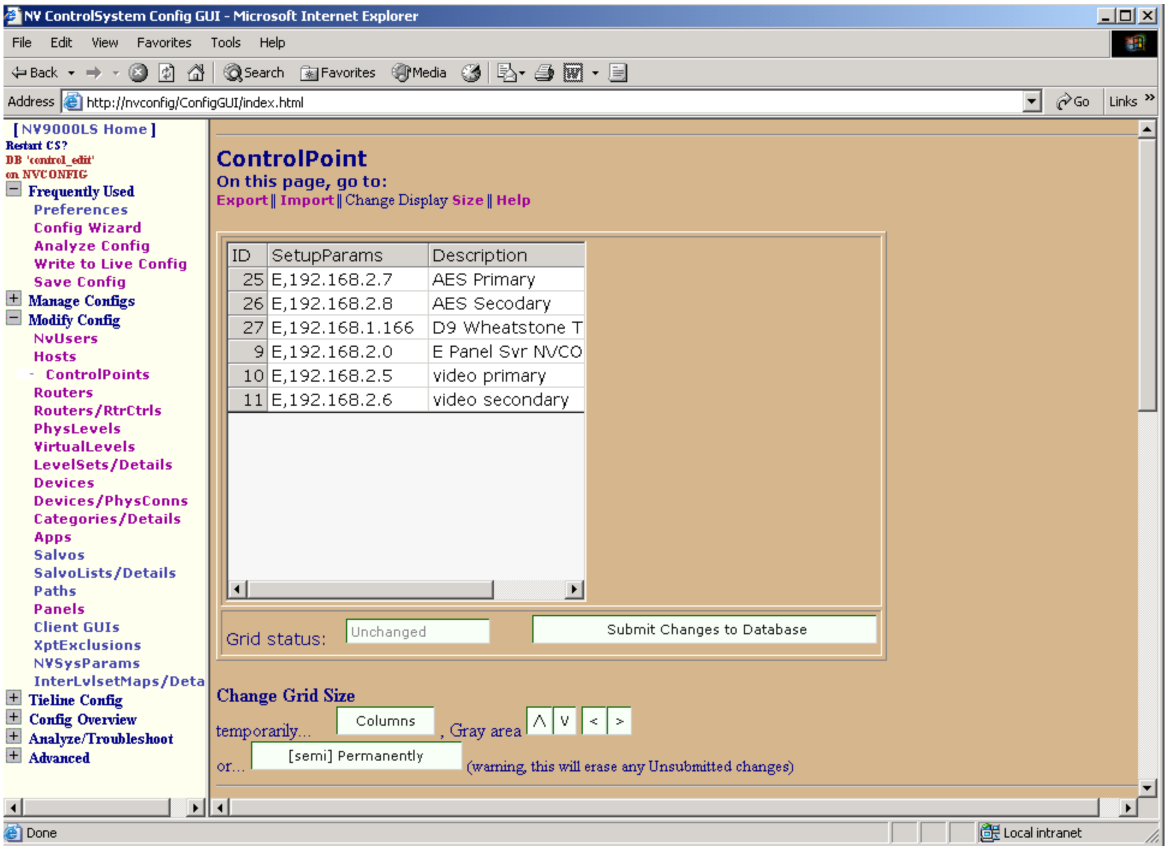
***UTAH 300 and 400* - SC3 or SC4 System Controller**

- Use the "Utah" Translator router type. Utah RCP3 Protocol.
- Utah requires that you map all signals to a control set.
- The SC3 needs to be configured with a UN object to enable the 0x12 interface that Translator is looking for.
- Use the RMS program. Select the 'SCP Tables' tab and then click the 'Create' button in the 'UN Object Status' box. Create the UN object with the same parameters and mapping as the PL object under the 'CSP Tables' tab.
- The SC4 uses a different configuration method. An alias cross reference table of Utah source signals and their levels is created, only the aliases signal ID is used. Please contact Utah support for details.

nVision 5128

- nVision UniConfig 3.0.0 emulator used for communications interface development.
- Requires a Control Point of the Type Ethernet be defined in the NVControlSystemGUI. Use the IP address of the PC running Translator.
- Requires creation of an APP in the NVControlSystemGUI pointing to the previously defined Control Point.
- Default TCP Port is 9193, IP address in Translator ini points to the Housenet NIC.

Sample NVControlSystemGUI screenshots:



ControlPoint

On this page, go to:

Export || Import || Change Display Size || Help

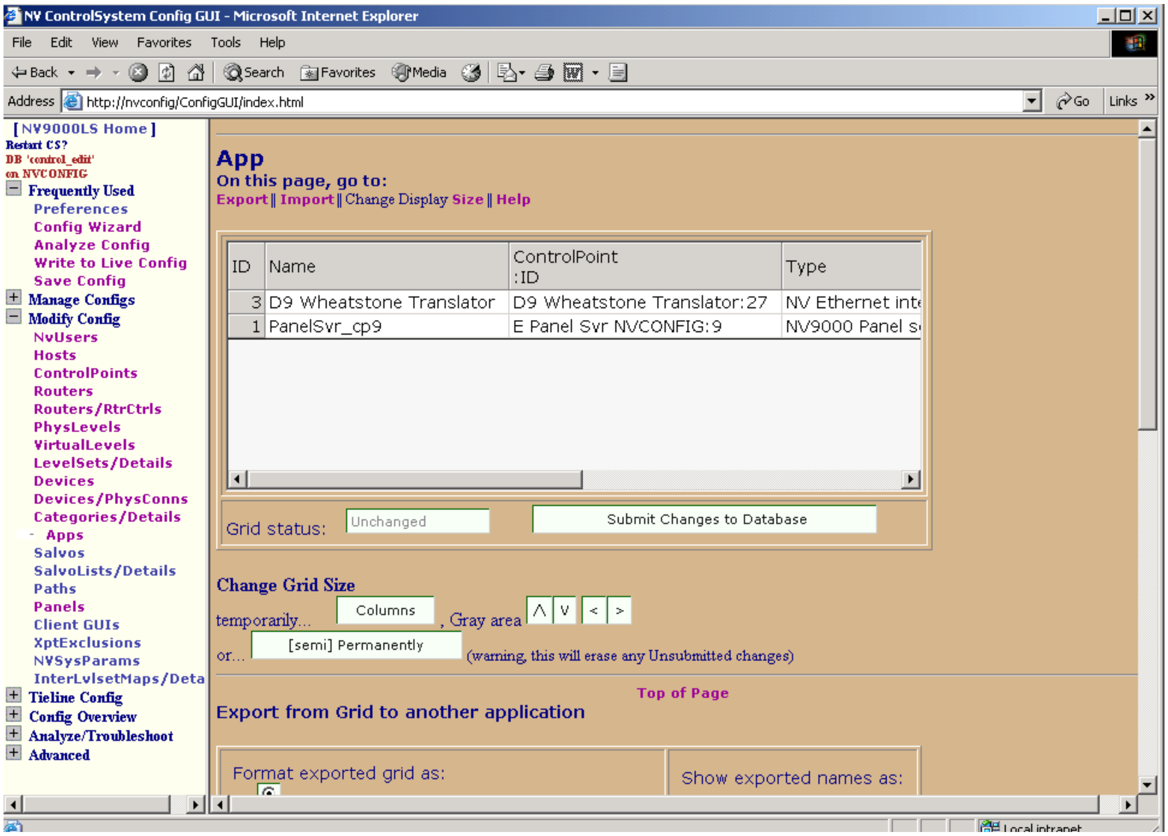
ID	SetupParams	Description
25	E,192.168.2.7	AES Primary
26	E,192.168.2.8	AES Secodary
27	E,192.168.1.166	D9 Wheatstone T
9	E,192.168.2.0	E Panel Svr NVCO
10	E,192.168.2.5	video primary
11	E,192.168.2.6	video secondary

Grid status: Unchanged

Change Grid Size

temporarily... Columns , Gray area

or... [semi] Permanently (warning, this will erase any Unsubmitted changes)



App

On this page, go to:

Export || Import || Change Display Size || Help

ID	Name	ControlPoint :ID	Type
3	D9 Wheatstone Translator	D9 Wheatstone Translator:27	NV Ethernet inte
1	PanelSvr_cp9	E Panel Svr NVCONFIG:9	NV9000 Panel s

Grid status: Unchanged

Change Grid Size

temporarily... Columns , Gray area

or... [semi] Permanently (warning, this will erase any Unsubmitted changes)

Export from Grid to another application

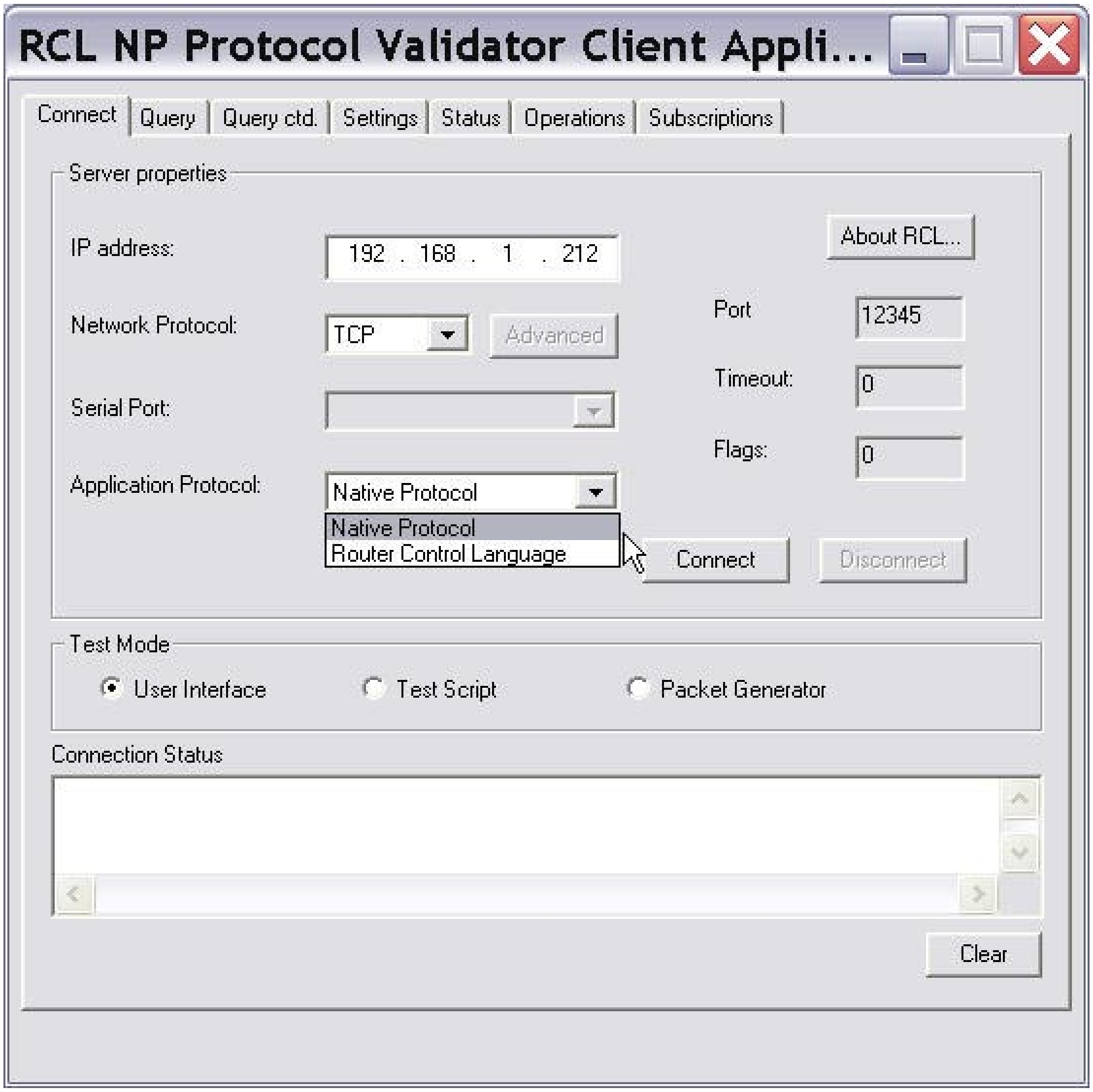
Format exported grid as:

Show exported names as:

Appendix A - Grass Valley Encore RCL Server Configuration

RCL Server configuration

- Use the Encore OUI to configure a Control Panel Server of the type RCL server.
- Enter the IP address of the Wheatstone PC running Translator. Set Flags to none, Refresh 0, Protocol Type to RCL.
- Note that Sources signal ID numbers are “zero” based (i.e signal 1 ID is 0, signal2 ID is1, signal3 ID is 2, etc.)
- Use the RCL - NP Validator program supplied by Grass Valley to check for a connection to Translator.



IP address: 192 . 168 . 1 . 212

Network Protocol: TCP

Serial Port:

Application Protocol: Native Protocol

Port 12345

Timeout: 0

Flags: 0

Test Mode

● User Interface ○ Test Script ○ Packet Generator

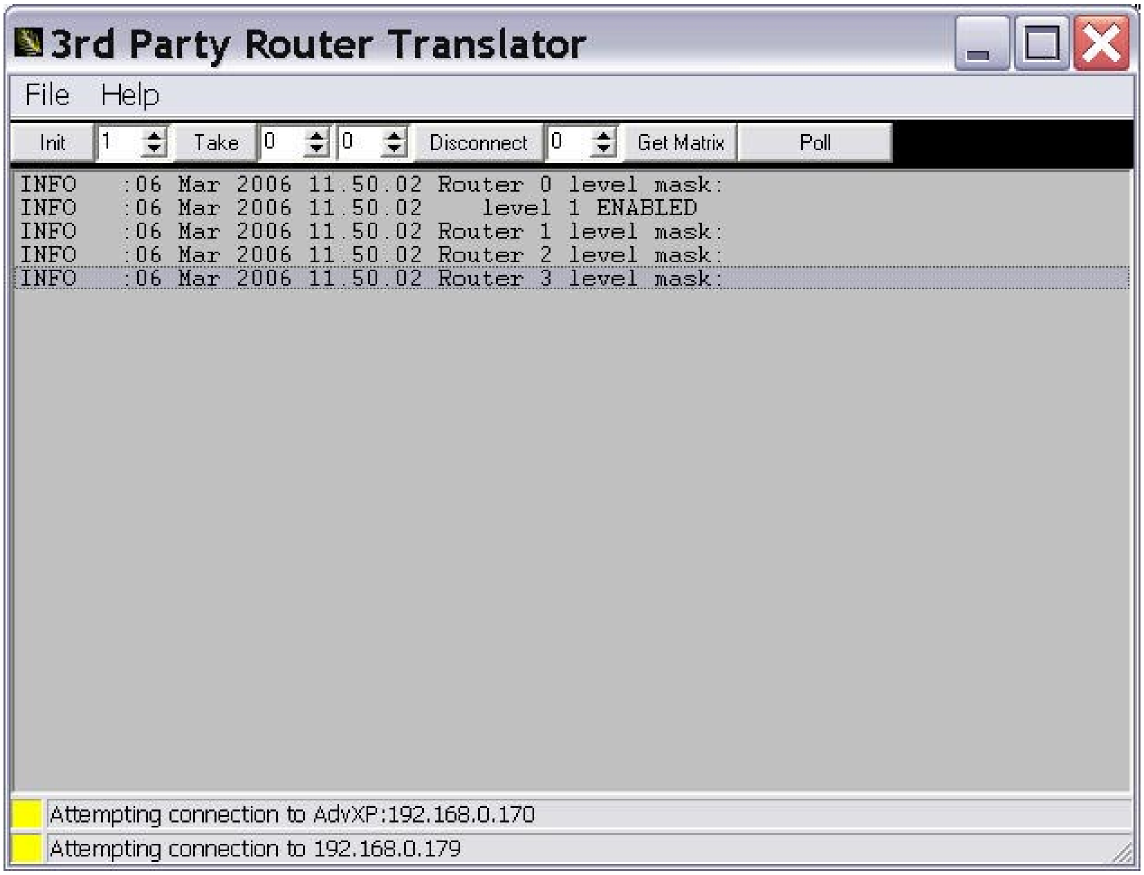
Connection Status

Appendix B - Using the Debug Bar in Translator

A rudimentary Debug Bar is available for testing communications with the house router. A Telnet or other type of debug session should be simultaneously opened on the house router to correlate Takes done from the debug bar with external communications received by the house router.

Manual Take

Enter the signal ID of the source and destinations then press the TAKE button.
Other buttons are for Translator software development and may or may not provide useful feedback.
If the Debug Bar is not visible you must edit the Translator INI file and set Debugbar=1.



```
INFO   :06 Mar 2006 11.50.02 Router 0 level mask:
INFO   :06 Mar 2006 11.50.02    level 1 ENABLED
INFO   :06 Mar 2006 11.50.02 Router 1 level mask:
INFO   :06 Mar 2006 11.50.02 Router 2 level mask:
INFO   :06 Mar 2006 11.50.02 Router 3 level mask:
```

Attempting connection to AdvXP:192.168.0.170

Attempting connection to 192.168.0.179