Utah Scientific

MX-Lator

with Global Routing

System Setup and Operations

MX-Lator Operations Guide

- Document Number: 82101-0105
- Document Version: 4.0
- Date: November 4, 2015
- Printed in U.S.A.

Copyrights and Trademarks

- MX-Lator is a trademark of Utah Scientific, Inc.
- Windows 2000, NT, and XP are registered trademarks of Microsoft Corporation.
- All other product names and any registered or unregistered trademarks mentioned in this guide are used for identification purposes only and remain the exclusive property of their respective owners.

Notice

Information contained in this guide is subject to change without notice or obligation. While every effort has been made to ensure that the information is accurate as of the publication date, Utah Scientific, Inc. assumes no liability for errors or omissions. In addition, Utah Scientific, Inc. assumes no responsibility for damages resulting from the use of this guide.

FCC Compliance (USA) and Digital Equipment Compliance (Canada)

This equipment has been tested and found to comply with the limits for a Class A, digital device, pursuant to Part 15, Subpart B of the FCC Rules and the Canadian EMC Requirement (ICES-003). These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio

communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case, the user will be required to correct the interference at their own expense. Shielded cables must be used to ensure compliance with the FCC Class A limits.

Declaration of Conformity

Utah Scientific, Inc.

4750 Wiley Post Way, Suite 150
Salt Lake City, Utah 84116-2878 U.S.A.

We declare our sole responsibility that the U-Con Control System is conformance with the following standards:

- EN50081-1 Generic Emission Standard
- EN50082-1 Generic Immunity Standard
- IEC-950 Product Safety
- C-UL 1950 Product Safety
- UL 1950 Product Safety

Following the provisions of the Directive(s) of the Council of the European Union:

- EMC Directive 89/336/EED
- Low Voltage Electrical Directive 72/23/EEC

Utah Scientific, Inc. hereby declares that the product specified above conforms to the above Directive(s) and Standard(s).

CE

Important Safeguards and Notices

This section provides important safety guidelines for the Operator and Service Personnel. Specific warnings and cautions are found throughout the guide where they apply, but may not appear here. Please read and follow the important safety information, specifically those instructions related to risk of fire, electric shock, or injury to persons.

Safety Symbols

- Hazardous Voltage symbol

- Caution symbol. The product is marked with this symbol when it is necessary to refer to the manual to prevent damage to the product.

Warnings

Please observe the following important warnings:

- Any instructions in this guide that require opening the chassis, changing a power supply, or removing a board, should be performed by qualified personnel only. To reduce the risk of electric shock, do not perform any service unless you are qualified to do so.
- Heed all warnings on the unit and in the operating instructions.
- Do not use this product in or near water. Disconnect AC power before installing any options or servicing the unit unless instructed to do so by this manual.

- This product is grounded through the power cord ground conductor. To avoid electric shock, plug the power cord into a properly wired receptacle before connecting the product inputs or outputs.
- Route power cords and other cables so they won't be damaged.
- The AC receptacle (socket) should be located near the equipment and be easily accessible.
- Disconnect power before cleaning. Do not use any liquid or aerosol cleaner - use only a damp cloth.

- Dangerous voltages exist at several points in this product. To avoid personal injury, do not touch exposed conductors and components while power is on. Do not insert anything into either of the systems two-power supply cavities with power connected.
- Do not wear hand jewelry or watches when troubleshooting high current circuits, such as power supplies. During installation, do not use the door handles or front panels to lift the equipment as they may open abruptly and injure you.
- To avoid fire hazard when replacing fuses, use only the specified correct type, voltage and current rating as referenced in the appropriate parts list for this product. Always refer fuse replacement to qualified service personnel.
- Have qualified personnel perform safety checks after any service.

Cautions

Please observe the following important cautions:

- When installing this equipment do not install power cords to building surfaces. To prevent damage when replacing fuses, locate and correct the problem that caused the fuse to blow, before reconnecting power.
- Use only specified replacement parts

Notices

Please observe the following important notes:

- When the adjacent symbol is indicated on the chassis, please refer to the manual for additional information.
- For the HD-2020 Chassis and Master Control Panel, refer to “Connecting and Disconnecting Power” - Chapter 2 (Hardware Installation).

Company Information

Utah Scientific, Incorporated

4750 Wiley Post Way, Suite 150
Salt Lake City, Utah 84116-2878 U.S.A.

- Telephone: +1 (801) 575-8801
- FAX: +1 (801) 537-3098
- Technical Services (voice): +1 (800) 447-7204
- Technical Services (FAX): +1 (801) 537-3069
- E-Mail -General Information: info@utsci.com
- E-Mail -Technical Services: service@utsci.com
- World Wide Web: http://www.utahscientific.com
- **After Hours Emergency**: +1 (800) 447-7204. Follow the menu instructions for Emergency Service.

Warranty Policies

Hardware Warranty

Utah Scientific, Inc. warrants to the original purchaser that the Utah Scientific hardware is free from defects in materials and workmanship and will perform substantially in accordance with the accompanying written materials under normal use and service for a period of ten (10) years from the date of shipment. Any implied warranties on hardware are limited to ten (10) years. Some states/jurisdictions do not allow limitations on duration of an implied warranty, so the above limitation may not apply to certain specific purchasers.

Software Warranty

Utah Scientific warrants that the software will perform substantially in accordance with the accompanying written materials for a period of one (1) year from the date of shipment.

Customer Remedies

For the first one (1) year after purchase of the software and the first ten (10) years after the date of purchase of the hardware, Utah Scientific's and its suppliers' entire liability and purchaser's exclusive remedy shall be, at Utah Scientific's option, either:

- Return of the price paid, or
- Repair or replacement of the software or hardware that does not meet the above warranties and is returned to Utah Scientific under the returned materials authorization (RMA) process with freight and forwarding charges paid.

After the initial warranty periods, purchaser's exclusive remedy is the repair or replacement of the hardware upon payment of a fixed fee to cover handling and service costs based on Utah Scientific's then-current price schedule. The above warranties are void if failure of the software or hardware has resulted from an accident, abuse, or misapplication. Any replacement software or hardware will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer.

No other warranties. To the maximum extent permitted by applicable law, Utah Scientific and its suppliers disclaim all other warranties, either express or implied, including, but not limited to implied warranties of merchantability and fitness for a particular purpose, with regard to the software, the accompanying written materials, and any accompanying hardware. This limited warranty gives the purchaser specific legal rights. These rights may vary in certain states/ jurisdictions.

No liability for consequential damages. To the maximum extent permitted by applicable law, in no event shall Utah Scientific or its suppliers be liable for any damages whatsoever (including without limitation, damages for loss of business profits, business interruption, loss of business information, or any other pecuniary loss) arising out of the use of or inability to use Utah Scientific products, even if Utah Scientific has been advised of the possibility of such damages. Because some states/jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply in those circumstances.

Table of Contents

MX-Lator Setup and Operations Guide

Chapter 1 - Setup and Operations

Chapter 2 - SC-400 Configuration

Chapter 3 - MX-Lator Configuration File

Chapter 4 - Operations

Chapter 1

1

MX-Lator Setup and Operations

This Section contains the following:

Introduction

The MX-Lator (Control Translation Unit) is designed to provide reliable integration of external routers within an SC-4/SC-400 system. This is accomplished by translating critical router commands within the system's MX-Bus architecture. When multiple MX-Bus router frames are used in a system, such as separate audio and video frames, they are connected to the MX-Lator frame by running MX-Bus cables to each frame in a daisy-chain connection.

The MX-Lator is capable of maintaining control of third party routers through one of its six RS-422 serial ports. System connections also include RJ-45 Ethernet, alarm, vertical interval signal reference, and timecode input.

- Ethernet port is used with UCP and SCP series control panels.
- SC-400 controller offers (2) serial ports for use with external devices such as automation controllers, Under Monitor Displays, etc.
- Looping sync input is provided for connection of a composite video vertical interval reference signal (NTSC and PAL).
- Alarm connector provided for remote connection of the SC-400's operating alarms. These typically report major internal faults such as power supply failures, internak temperatire alarms, and controller failures.

General Overview

How the System is Used

There are three different configurations that are possibe with the MX-Lator:

1. **MX-Lator ONLY**

When only MX-Lator cards are installed in the chassis,

the system is used with an external SC-4 controller to control legacy USI routers or third party router equipment.

The MX-Lator is a device intended to translate the USI proprietary MX-Bus to other USI router protocols and to third party interfaces.

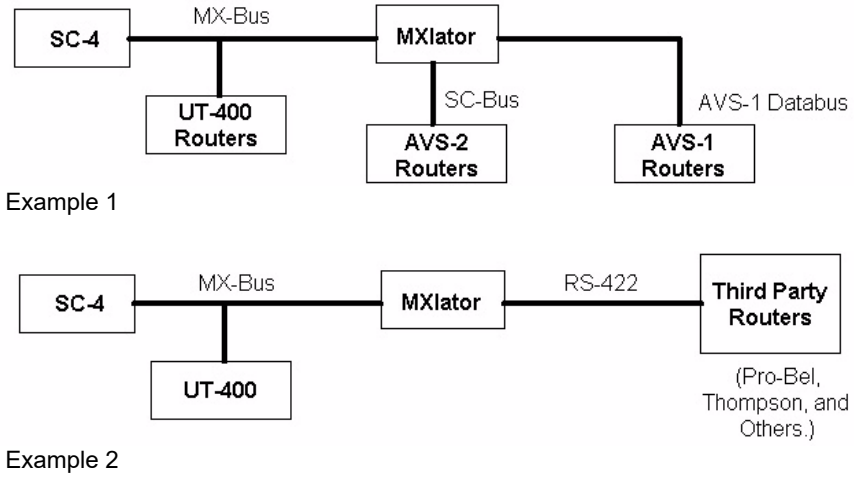


Figure 1-1. System Layout Example 1

The MX-Lator receives all matrix information from the MX-Bus. It has a local sync input that allows for vertical internal switching of local routers.

This arrangement allows for standard operation of the SC-4 controller while all other router control tasks are off-loaded to the MX-Lator.

2. SC-400 ONLY

When only SC-400 cards are installed, the MX-Lator chassis is used as a cost and feature reduced router controller. Router matrix size is limited to 528x528 and 8 levels, while panel types are limited to UCP or SCP types.

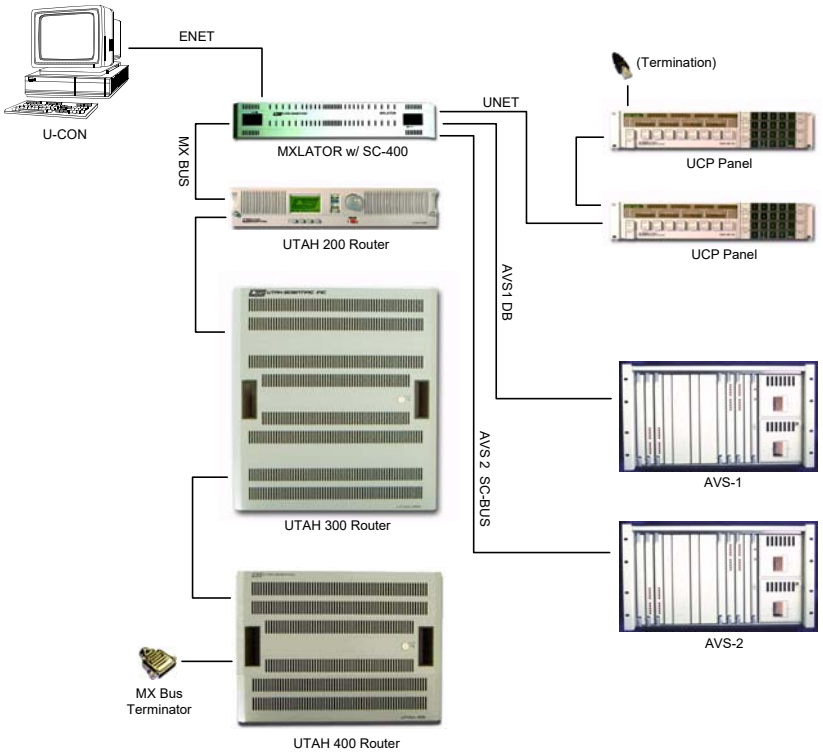


Figure 1-2. System Layout Example 2

3. Combined MX-Lator and SC-400 System

This configuration is for systems with legacy USI routers or third party routers that do not need all the functionality of an SC-4. Matrix size is limited to 528x528 by eight levels.

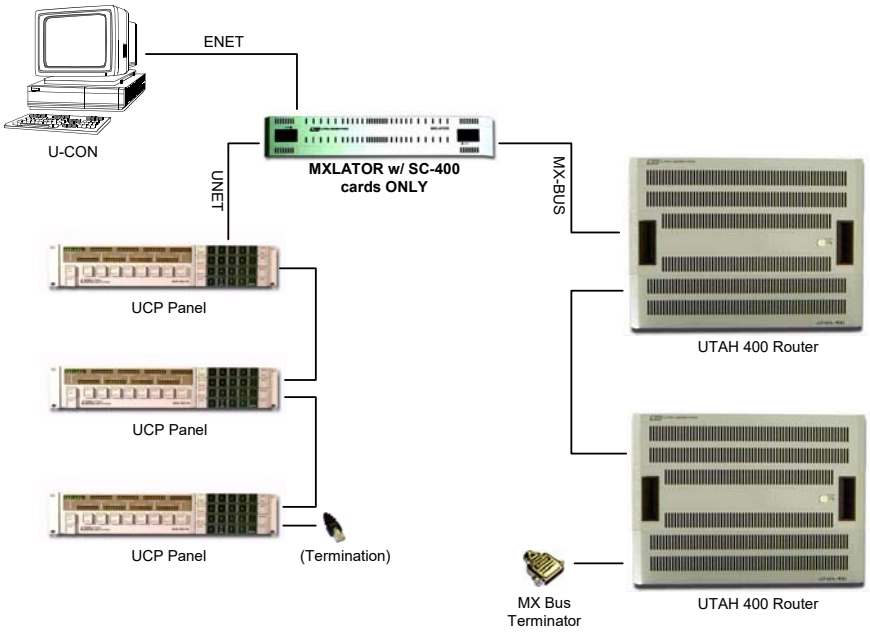


Figure 1-3. System Layout Example 3

Connections to routers in this configuration are dependent on the system. MX-Bus, AVS-1, or AVS-2 can all be controlled, in any combination.

System Layout

- Attachments – MX-Lator rear panel
- Mx-Lator board layout for control / indicator locations
- MX-Lator configuration file example

The MX-Lator product is used to Translate MX Bus commands to AVS-1 databus, AVS2 SC bus, or a variety of third party router control systems from other manufacturers, such as Grass Valley or Thompson. (The MX bus is the current router crosspoint control mechanism for USI routing switchers.) It's main purpose is to allow the control of a wide variety of routing switchers, both current and legacy product, from a single control system.

The MX-Lator is housed in a 2RU enclosure that contains a redundant pair of MX-Lator cards (USI PN 121161-1), dual redundant power supplies (140080-01) and, optionally, a redundant pair of SC400 control cards. In a small, limited system, the SC400 cards could be the system controller. In larger systems, an external SC4 controller will be the system controller.

The MX-Lator functionality is determined by programming entered into a configuration file. This is an extensive document that defines which segments of the Mix-ups translate into which MX-Lator control bus.

Receiving the System - Unpacking

When you receive your MX-Lator system, inspect the shipping carton for signs of damage. Contact your dealer and shipper immediately if you suspect any damage has occurred during shipping. Check the contents of the carton against your Utah Scientific order and verify it against the shipping manifest. If any items are missing, contact your dealer or Utah Scientific immediately.

Save the shipping box and material for future use, in case the unit may have to be shipped back to Utah Scientific.

> ***Note: The MX-Lator weighs approximately eighteen pounds; with shiping materials and accessories, the box may weight more than thirty pounds.***

The unit is wrapped in anti-static plastic prior to boxing up.

Equipment List

- 2 Rack Unit MX-Lator Frame
- (2) Power Cords

Chassis Loading

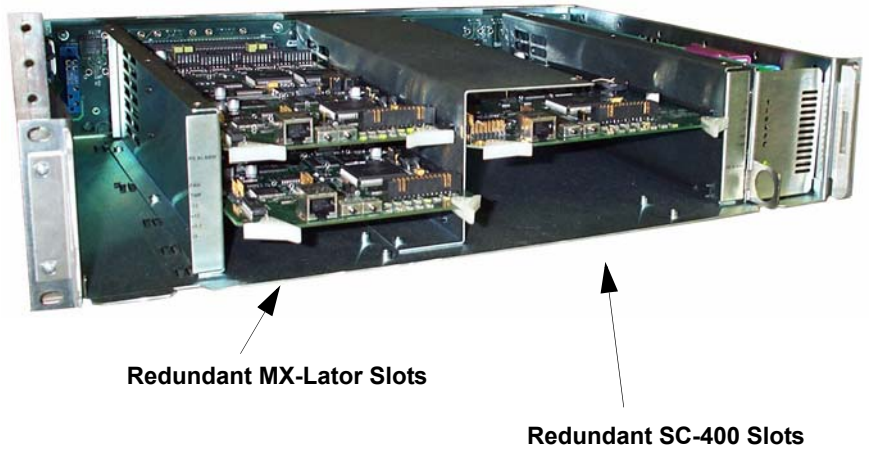


Figure 1-4. Slot Indications

System Installation

Rack Mounting

Mount the MX-Lator in a standard 19 inch equipment rack. It should be located near the routers with which it will be interfaced.

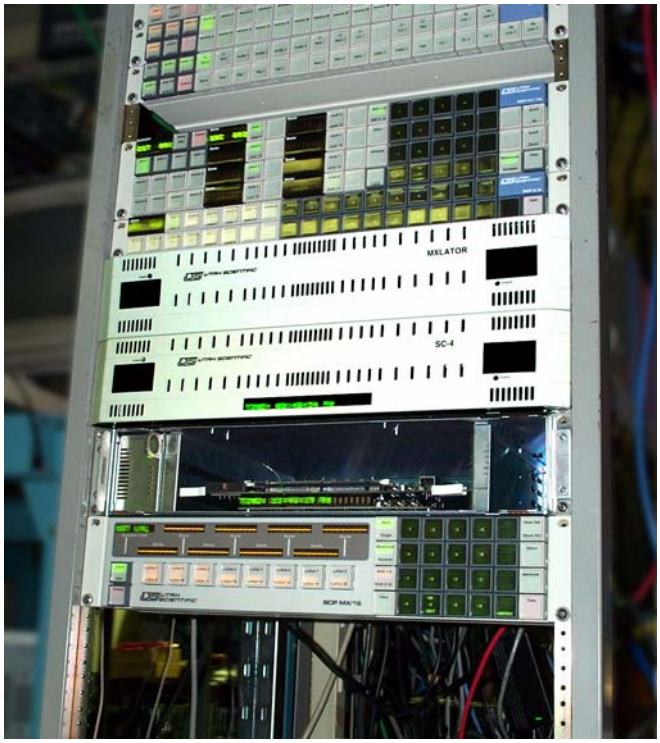


Figure 1-5. MX-Lator rack mounting

Rear Panel Connection Information

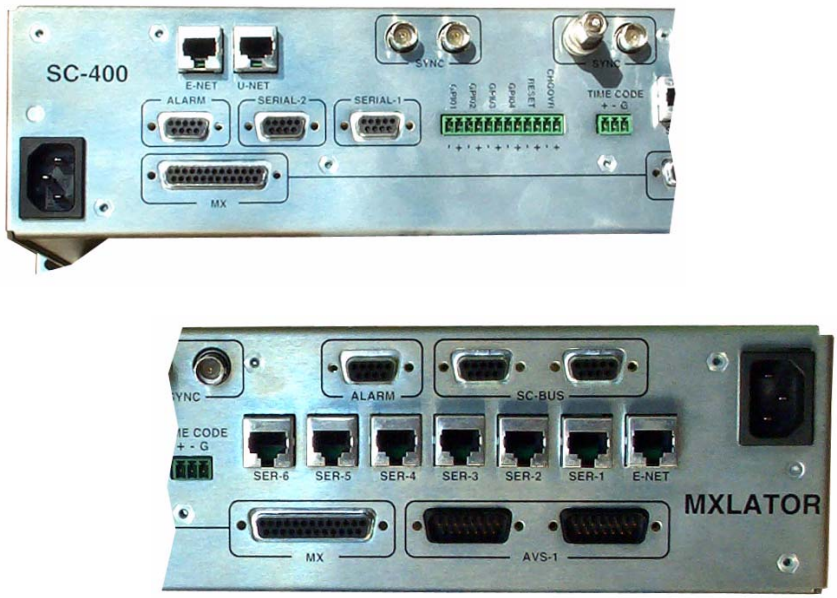


Figure 1-6. MX-Lator Rear views

MX-Lator

(2) SC-Bus ports

Connects to USI AVS-2 and DDS-2 routers via standard CC-2 cables. This bus connection should be terminated at each end with an AVS-2 terminators. If no AVS-2 routers are in the system, no connection to these ports are required.

SYNC loop Through

If any of the routers controlled by the MX-Lator are desired to switch in vertical interval, analog black burst in the proper time domain should be connected to this port. The port is a passive loop-through, so it should be terminated in 75 ohms at it's end, whether that be at the MX-Lator chassis or somewhere else downstream.

(2) AVS-1 Ports

This port connects to AVS-1B frames through a standard DC-1 data cable. No termination is placed on this bus.

Alarm Port

This Female 9 pin D-SUB connector is used by the MX-Lator to communicate errors to the user. It contains a SMPTE RP-159 compatible contact closure and four TTL alarm lines. It's pinout is as follows –

i. Pins 6,7,8 –Ground.

ii. Pin 1 – TTL Alarm Out 1.

iii. Pin 2 – TTL Alarm Out 2.

iv. Pin 3 – TTL Alarm Out 3.

v. Pin 4 – TTL Alarm Out 4.

vi. Pins 5, 9 – SMPTE Alarm Contact Closure.

E-NET RJ-45 connector

This is a standard 10/100 Ethernet port for connection to configuration computers, SC-4 or SC-400 controllers.

SER-1 through SER-6

These ports are RS-232 or RS-422 switchable ports that can be configured as needed to control third party routing switcher systems. Their Pinout is-

In RS-232 Mode –

1. Pin 1 – Transmit out.
2. Pin 2 – DTR out.
3. Pin 3 – RX in.
4. Pin 4 – DSR in.
5. Pin 5 – RTS out.
6. Pin 6 – Ground.
7. Pin 7 – CTS in.
8. Pin 8 – Ground.

ii. In RS-422 Mode –

1. Pin 1 – TX- out.
2. Pin 2 - Ground.
3. Pin 3 – RX+ in.

4. Pin 4 – Ground.
5. Pin 5 – TX+ out.
6. Pin 6 – Ground.
7. Pin 7 RX- in.
8. Pin 8 – Ground.

Timecode port

i. This three position Terminal clock is an input for Longitudinal Time Code. Insert the non-inverted leg of the timecode differential pair into the '+' pin, the inverted leg into the '-' pin, and shield to the 'G' pin.

GPIO Ports

Four Bi-Directional GPIO's are available for user assigned functions.

i. When configured as an output, the will provide a dry contact closure across the two pins when activated.

ii. When configured as an input, the user must provide a voltage from between 5-12V across the + and – pins to activate the input.

Common Ports

MX Bus loop through

The MX Bus is a router control bus that connects via 25 pin cable from controllers (SC4, SC400) to router frames. Regardless as to which products are installed in this chassis, the SC400, the MX-Lator or both, MX Bus must connect to the two 25 Pin Female connectors on this chassis and any MXBus based routers to be controlled. MX Bus must be terminated to operate correctly. USI provided terminators must be installed on the unused ports at the last piece of equipment on each end of the MX Bus.

Remote Changeover and Remote Reset

i. Remote Changeover is used to switch the Active card in the redundant pair to standby, and activate the redundant card. To initiate a remote changeover, the user must provide a voltage from between 5-12V across the + and – pins.

ii. Remote Reset is used to reset all of the control boards in the chassis and force them to restart operation from scratch. To activate remote reset, the user must provide a voltage from between 5-12V across the + and – pins.

Card based controls and indicators

MX-Lator

i. PCB Based Indicators.

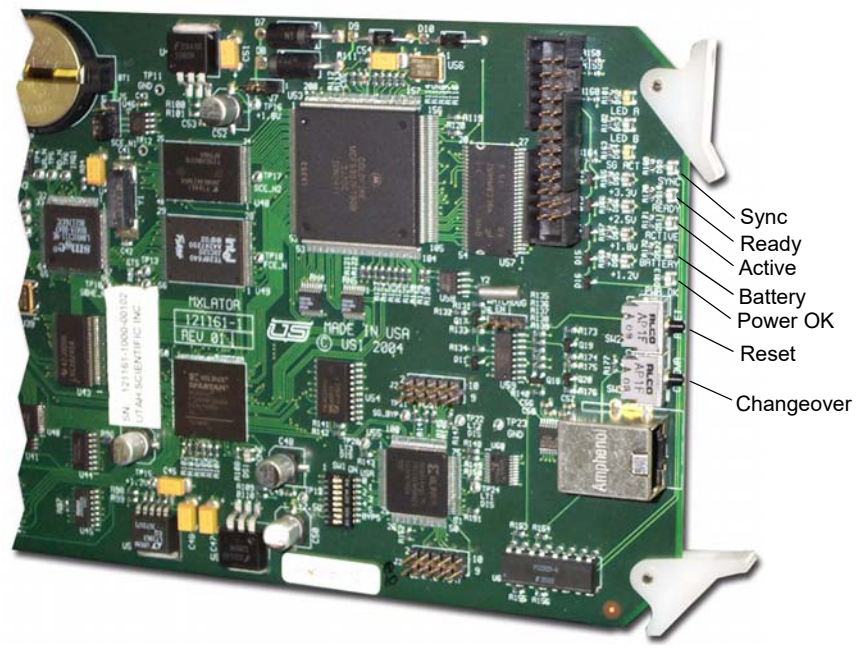


Figure 1-7. MX-Lator Indicators

Red Voltage Alarm LED's

a. DS8 – 5V supply error if on.

b. DS7 – 1.2V or 1.8V supply error if on.

c. DS6 – 2.5V supply error if on.

d. DS5 – 3.3V supply error / RESET active if on.

e. DS12 – Battery low error if on.

Power OK LED green DS13. If this LED is illuminated, all board power supplies are in tolerance.

Activity Indicators

a. DS9 green SYNC LED. Indicates that SYNC is applied if on.

b. DS4 Scangate Active yellow LED. Indicates that the Scangate part is selected if on.

c. LEDB green Ethernet Link LED. Indicates that the Ethernet port has negotiates link with the connected switch or hub.

d. LEDA yellow Ethernet activity LED. Lights when a TX or RX packet is processed by the Ethernet part.

Board state LED's

a. DS11 green ACTIVE LED. Indicates that this board is the 'master' board in the redundant pair.

b. DS10 green READY LED. Indicates that the two cards in a redundant pair are in sync and that a changeover can be performed if requested.

ii. PCB Based Controls.

1. Dipswitch SW1

a. S1 – Do not run Linux if ON.

b. S2 – Use partition 1 if OFF, partition 2 if ON.

c. S3 – No Function

d. S4 – No Function.

e. S5 – No Function.

f. S6 – Start Linux, but do not start application if ON.

g. S7 – Place Scangate part in Bypass if ON.

h. Normal operation has all of these switches set to OFF.

2. Battery Backed RAM default Jumper J5

a. 1-2 Normal Operation. 2-3 Remove power from Battery Backed RAM.

b. If the user wishes to default BB RAM, remove the board from the chassis, place J5 between pins 2-3, wait 1 minute, place J5 between 1-2, and re-install card.

3. CPU Core Voltage Select, J7

a. Leave between pins 1-2 ALWAYS.

4. Watchdog Enable/Disable jumper, J6

a. 1-2 Enables watchdog, 2-3 disables watchdog.

b. This jumper should always be between pins 1-2 in normal operation.

5. SW2 Changeover

a. Press this momentary switch on the card which has it's ACTIVE LED illuminated to change control to the standby card. Note that this switch is only effective if two cards are installed, and the 'READY' LED is illuminated on each card.

6. SW3 RESET

a. Pressing this switch causes a full hardware reset of this card.

7. JP1-JP6 Serial Port Format select

a. These gang jumpers allow selection of either RS-232 or RS-422 for each of the six Third Party Serial ports. For proper operation, ensure that jumpers on both cards of a redundant pair are set the same way.

b. Power Supplies.

i. The power supplies are calibrated at the factory. No field adjustment is necessary.

ii. The supplies have Red LED's that indicate failure of the 5V, 3.3V, +-12V, integral fan and over temperature. They also have a bi-color LED that is green when all other failure leds are off, and red when one or more of the other alarms trip.

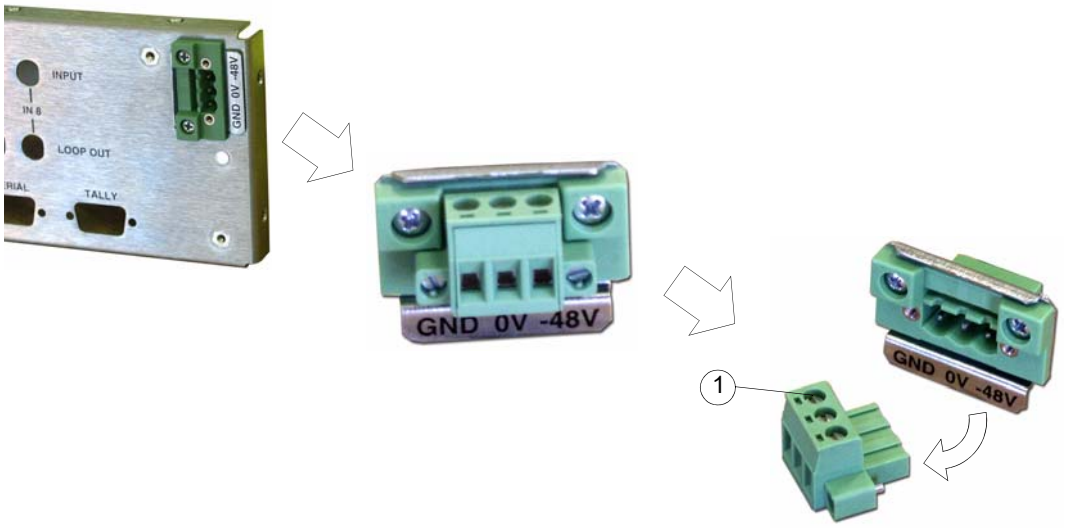
Switching from Active to Inactive Board

Important Note: To switch control from one board to the other, you must hold the reset button down for at least 5 seconds beyond the time the new board's activity light starts flashing.

DC Power Connectivity

The DC input at the rear of the chassis is noticeably different than its AC counterpart. The connection consists of three separate terminals:

- Ground - Frame or chassis grounding point
- 0V - Most positive leg of -48V DC connection.
- -48V - Most negative leg of -48V DC connection.



Note that this configuration is a DC-I or DC isolated connection.

The terminal strip is a small bracket containing three screws (see 1). Loosen the screws to remove the terminal from the back. This will expose the strip of wire (aprox. 1/4 of an inch).

Proper wire insertion into the removable terminal block

- Turn the screws counter clockwise to allow wire insertion (3 screws on block top).
- Strip 1/4” of the insulation from the new wires.
- Insert wire, then turn screw clockwise to tighten

Use 12 AWG wire (maximum)

The maximum branch circuit protection rating for the circuit feeding the SC-4 is 2 Amps

2

Chapter 2

SC-400 Configuration

Rear Panel Connection

SYNC loop through

If any of the routers controlled by the SC400 are desired to switch in vertical interval, analog black burst in the proper time domain should be connected to this port. The port is a passive loop-through, so it should be terminated in 75 ohms at it's end, whether that be at the MX-Lator chassis or somewhere else downstream. This sync can be the same, or different than the sync applied to the MX-Lator side of the chassis.

E-NET RJ-45 connector

This is a standard 10/100 Ethernet port for connection to configuration computers, Ethernet based control panels or MX-Lator assemblies.

U-NET RJ-45 connector

This connector is used to connect to USI SCP or UCP U-Net based control and status panels.

Alarm DB-9 Connector

This Female 9 pin D-SUB connector is used by the SC400 to communicate errors to the user. It contains four TTL alarm lines. It's pinout is as follows –

i. Pins 6,7,8 –Ground.

ii. Pin 1 – TTL Alarm Out 1.
iii. Pin 2 – TTL Alarm Out 2.
iv. Pin 3 – TTL Alarm Out 3.
v. Pin 4 – TTL Alarm Out 4.

Serial 1 and Serial 2

Switchable RS232 or RS422 serial ports for connection to automation systems or control computers. Format selection is made on the SC400 main card. Their pinouts are as follows

RS-232 Mode –

1. Pin 1 – RI Input.
2. Pin 2 – TX Out
3. Pin 3 – RX Input
4. Pin 4 – DSR Input.
5. Pin 5 – Ground
6. Pin 6 – DTR Out.
7. Pin 7 – CTS Input.
8. Pin 8 – RTS Output.
9. Pin 9 – CD Input.

ii. RS-422 Mode –

1. Pin 1, 4, 5, 6, 9 Ground
2. Pin 2 – TX- out.
3. Pin 3 – RX+ in.
4. Pin 7 – TX+ o ut.
5. Pin 8 – RX – in.

SC400

The SC-400 Control Board is used within the UTAH-400 64x chassis and serves as a functionality and part upgrade to the SC-200. Either slot can be used within the chassis, and when in place, the 64x64 Router becomes a self-contained unit.

Board Placement within the 64x64 router

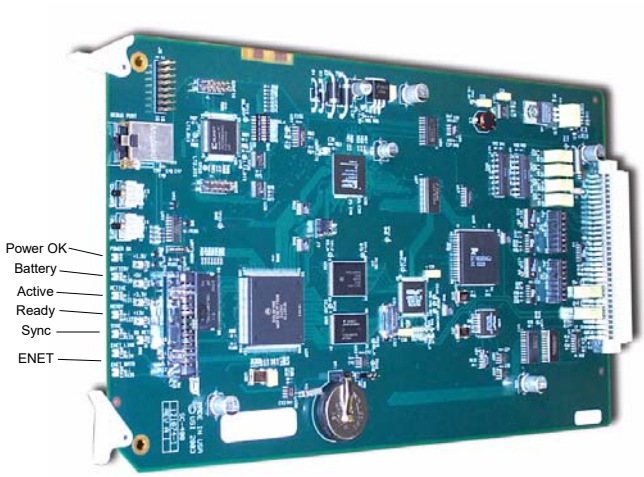


Figure 2-1. SC-400 Control Board View

The SC-400 LED Indications

- The ‘Ready’ LED indicates whether or not the cards are in sync and communicating with one other.
- Active, which indicates the ‘controlling’ card when the board is part of a redundant pair. The LED is green (ready) when the card is in default status. The current default card will give up control to its alternate when the **changeover** button is pressed.

- Power Ok. All power supplies are functional.
- Power Error indications. Located behind the Power Ok LED, these will illuminate to indicate a problem with a specified power supply.
- Battery Error. Will illuminate when the battery is low.
- Sync LED -- Will illuminate when an NTSC composite signal is in place -- allowing vertical interval switching.
- Ethernet link -- Illuminates when the ethernet port is in use.
- Note that the +12V Red LED will be on when this card is installed in the MX-Lator chassis. This is normal, and not a fault indication.

SC-400 Jumper Settings

- J2 Switch -- Used to clear the RAM
- JP1 -- Serial Port 1 -- **RS-232**
- JP2 -- Serial Port 2 -- **RS-422**

SC-400 Dipswitch Setting

User Switch 0

When set, the system will initiate a software load, though the Linux application itself will not start. This is typically used for factory troubleshooting.

User Switch 1

User switch 1 determines which of the two memory banks is in use. A change to either memory partition (0 or 1) will not affect the other. The actual switch position determines which memory partition is in use.

User Switch 5

Used to start the Linux operating system, but not the actual application.

User Switches 2 - 4

Undefined at this time.

Bypass Switch

Also used for factory troubleshooting. The default configuration is all switches positioned toward the silkscreened numbers, or to the *right* as the board is viewed from the front.

Configuring the MXLator

Chapter 3

Overview

There are various routers and systems that can either be controlled by the Utah Scientific control system, (which is most common), or that can control the Utah Scientific routers. The interface unit that is required for these applications is the MXLator, which is the translator between these various protocols.

Note: There is no GUI available to perform the setup for each of these configurations and therefore in order to setup the desired application this will need to be done manually in the configuration file.

There is one section in the configuration file titled [MX_BUS_DATA_START] that must be setup for most configurations. As a rule, any section that has the USI_SOURCE set to MX will require this section. This is the section that is used to convert the desired protocol into the Utah MX bus. The SC4/400 controller connects to the MX-Bus port on the MXLator unit and regardless of who is the master and who is the slave, this section defines which levels that are set up in the SC4/400 will be switched. This is the section that is used to convert the desired protocol into the Utah MX Bus protocol.

Current Applications

- Utah AVS-1 – DATA BUS (Controlled by Utah)
- Utah AVS-2 – SC BUS (Controlled by Utah)
- Thompson GVG - ES BUS and SMS-7000 (Controlled by Utah and Controlling Utah)
- Datatek – DATATEK BUS (Controlled by Utah and controlling Utah – also called bridge)
- Sandar – PROSAN BUS (Controlling Utah)
- Sandar – HPC BUS (Controlled by Utah and Controlling Utah)
- Probel – PROBELNS BUS (Controlled by Utah and Controlling Utah)
- Sierra Video – SIERRA BUS (Controlled by Utah)
- PESA 3500 – (Controlled by Utah)
- Leitch – LEITCH BUS (Controlled by Utah)
- Lawo – LAWO BUS (Controlled by Utah)

Programming: SC4/SC400/SCX400

In both cases where the Utah control system will be controlling another external unit as the slave or if it will be controlled by the external unit, the router and level mapping tables need to be configured as the first step in setting up the MXLator. Use the following steps to set up and program the SC4 configuration using UCON.

1. Locate and run the UCON program which is generally found in the C:\usi\Ucon directory. A shortcut may be found on the desktop in a folder called 'Utah Shortcuts'. You will see the following program displayed in figure 1.

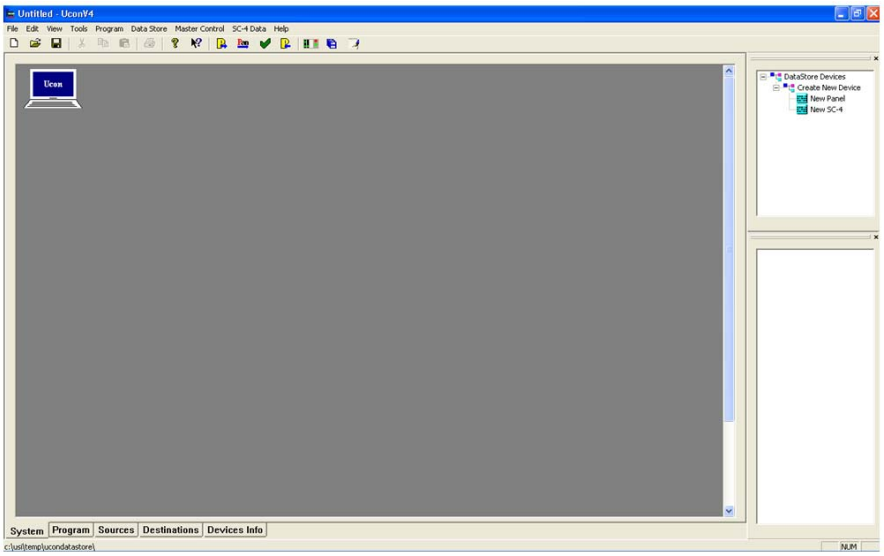


Figure 3-1.

2. Locate the Ucon system datastore and open it, which will look similar to figure 2 below after opened.

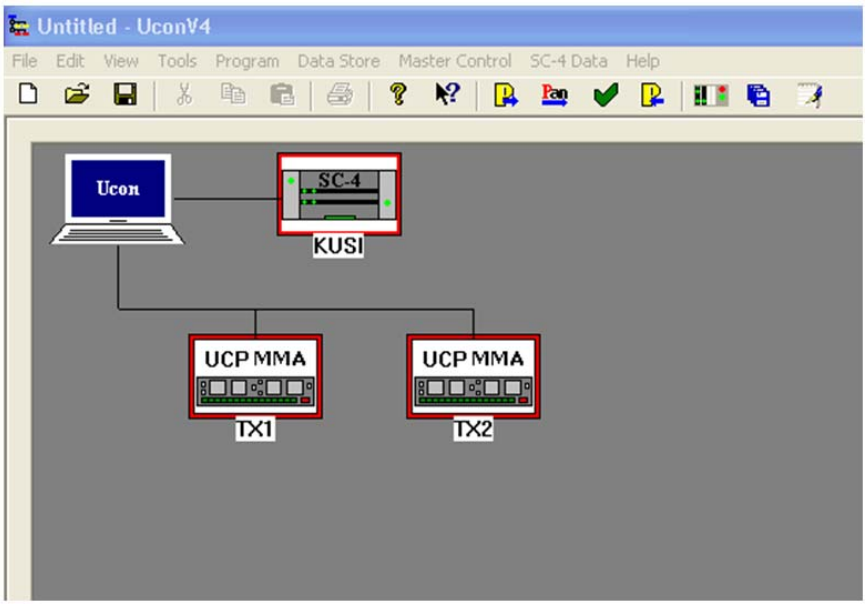


Figure 3-2.

3. Right click on the SC4 icon and select ‘Configure’ from the menu to see the following tables in figure 3.

Note: The following table is an example starting with a blank table. The MXLator levels will be added as new routers and levels in additions to your existing system. Physical 'Router' levels (top table) and virtual 'Levels' (bottom table) are related to Utah MX levels only. These routers and levels must be unique and can be any number. They are not related at all to the external system that will be attached to the MXLator. However, these numbers will be used when configuring the MXLator in its configuration file.

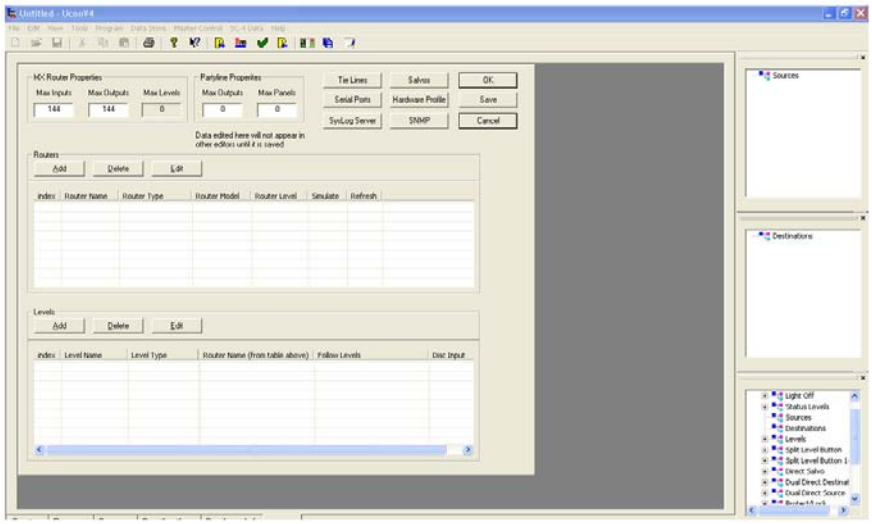


Figure 3-3.

4. You must first add the new router level in the top table and then associate the virtual level to that router in the bottom table. Click the 'Add' table in the top 'Routers' table to see the router definition table seen below in figure 4.

Figure 3-4.

5. Type the router name for the new external router that will work via the MXLator. Example below (figure 5) is just called ‘New Router’ but this can be any desired name and will only appear in this table. Next drop down the menu list under ‘Router Type’ (figure 6) and select the type of format for this new external router that will work via the MXLator.

Figure 3-5.

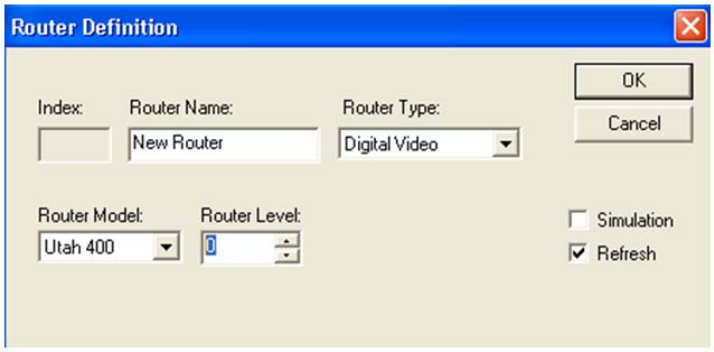


Figure 3-6.

6. Drop down the menu list under 'Router Model' (figure 7) and select the router model as either UT300, which is for AVS router types, or UT400, which is for all other external routers that will work via the MXLator. Then put in the physical level for this new external router in the 'Router Level' (figure 8) area. **NOTE**: This level begins with zero which represents the dipswitch setting of the router. Level 0 is the first level and its dipswitch would be set to zero. Put in the physical number for the router to be used and click OK on this screen after all the router information is added.

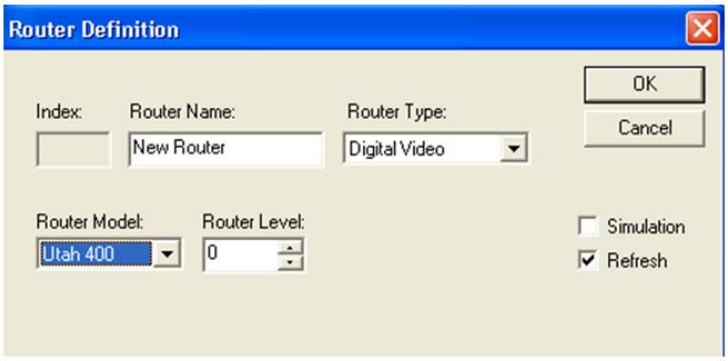


Figure 3-7.

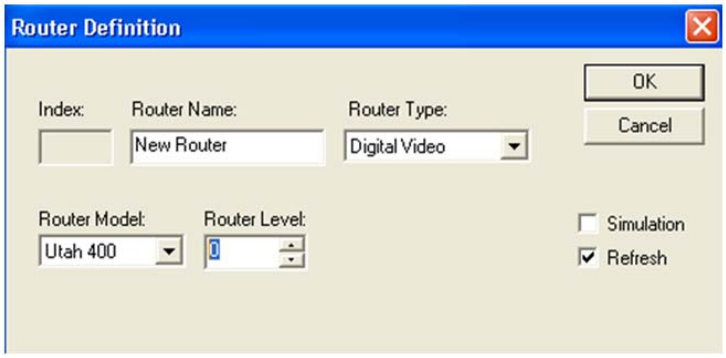


Figure 3-8.

7. With the data entered in the 'Routers' table now click on the 'Add' button in the 'Level' table at the bottom of the screen as seen in figure 9 below.

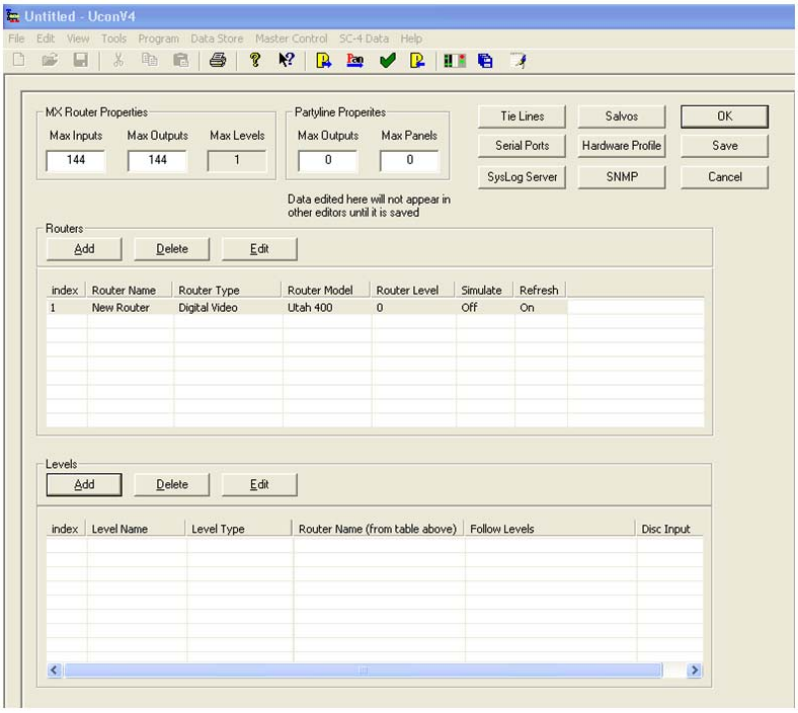


Figure 3-9.

8. Type the desired name in the ‘Level Name’ window (figure 10), which will appear in the source and destination table level columns. This can be any desired name. Minimum should be at least 5 characters for better clarity. Then drop down the ‘Level Type’ menu (figure 11) and select the same type that was chosen in the router table above.

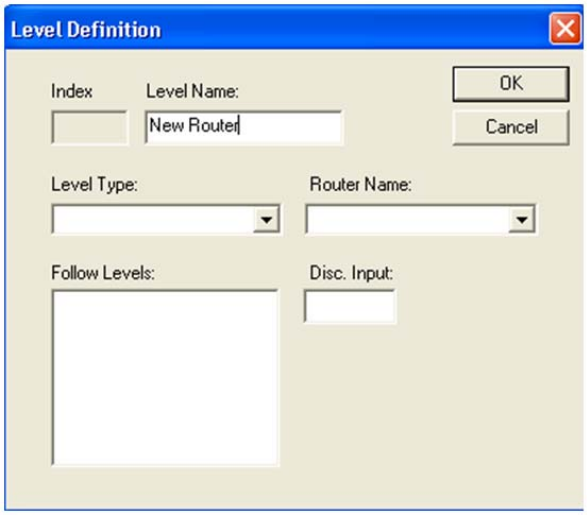


Figure 3-10.

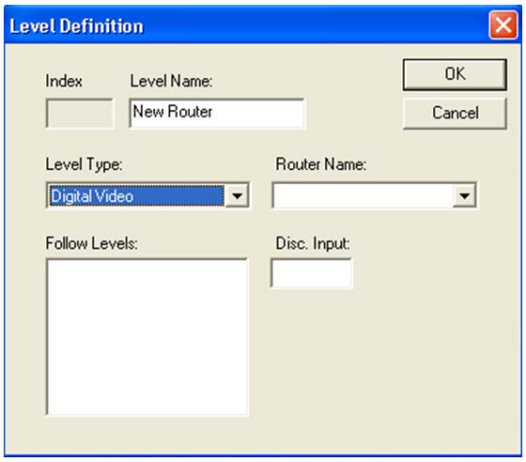


Figure 3-11.

9. Drop down the ‘Router Name’ menu (figure 12) and select the name of the router that was created in the router table above. This is what associates the virtual level to that physical router. (Nothing needs to be entered in the Follow Levels). Type the number 4095 in the ‘Dist’ window for internal use and then click OK after completing the Level Definition table.

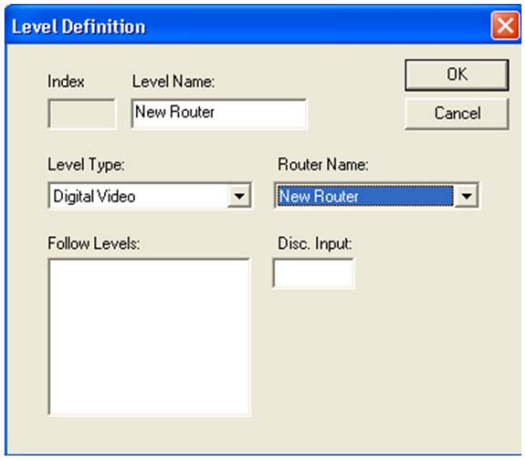


Figure 3-12.

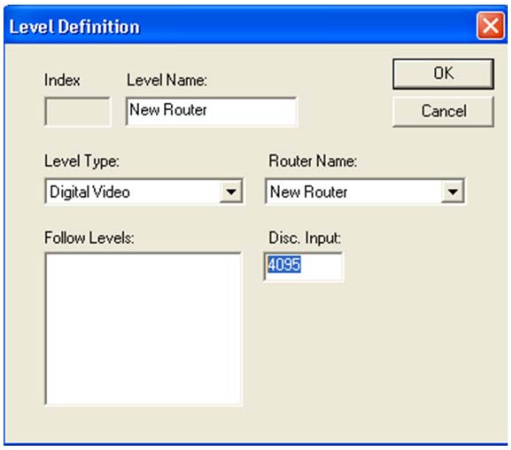


Figure 3-13.

10. Save your work by either clicking on the save icon in the upper menu or click ‘File’ then ‘Save’ in the upper left corner.

11. Click on the ‘Sources’ tab at the bottom of the screen which will open the source table (figure 14). Notice the top window bar has the name of the of the newly created router level (figure 15). This will appear in the level column that corresponds to the level number that was created in steps 7-9 above.

Figure 3-14.

Figure 3-15.

12. To add new sources begin by clicking on the ‘Add Device’ button to the right of the main source table. The following table (figure 16) will appear for adding a source device. **NOTE**: The destination table will appear exactly the same. Follow these same steps to add the destinations.

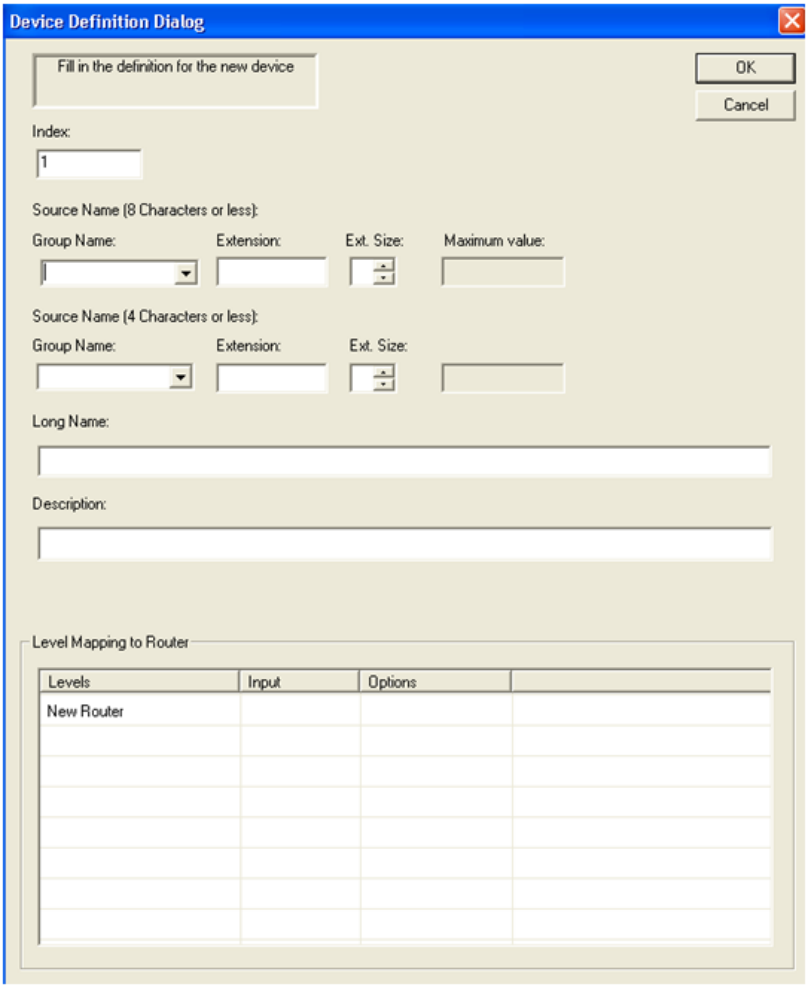


Figure 3-16.

13. Type the name of the group in the ‘Group Name’ window and the number in the ‘Extension’ window for that device in both the 8 and 4 character windows (Figure 17). Example would be if there were 3 VTR’s then the 8 character group would be called VTR and the extensions would be 1, 2 and 3 and the 4 character would be VT and its extensions would also be 1, 2 and 3. NOTE: The group and extension can only total 8 and 4 characters respective to their entries. So if the group name needs to be 7 entries (8 character window) then the extension can only be one entry of 0-9. And the same with the 4 character window. The length of the name will determine how many extensions or uses of that group can be made. The smaller the group name is the more of that group type can be used by increasing the ‘Ext Size’ box next to the name entry.

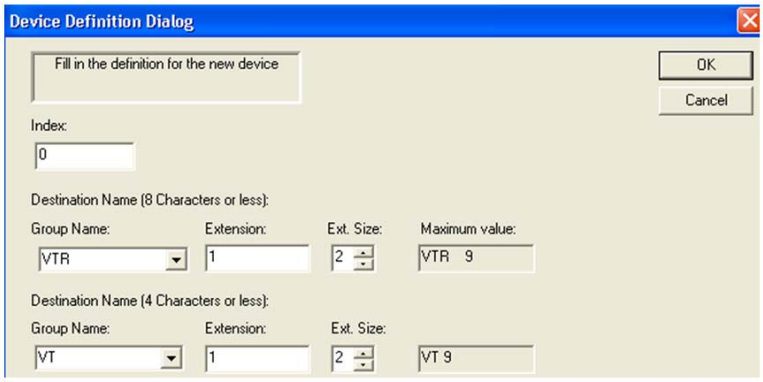


Figure 3-17.

14. Type in the number of the physical input or output number (depending which table, source or destination, you are working with) for that router device in the 'Level Mapping to Router' table (figure 18). The name of the level will be displayed in the left of the table and the entry is to be typed in the box below the 'Input' or 'Output' column depending which table you are working with. **NOTE**: the attribute and option tables are specific to Utah Scientific router types which are data router and UT300 routers as well as tie line configuration between router levels. These will generally never be used. Click OK when the table is complete for this device. Add all input and output devices before proceeding.

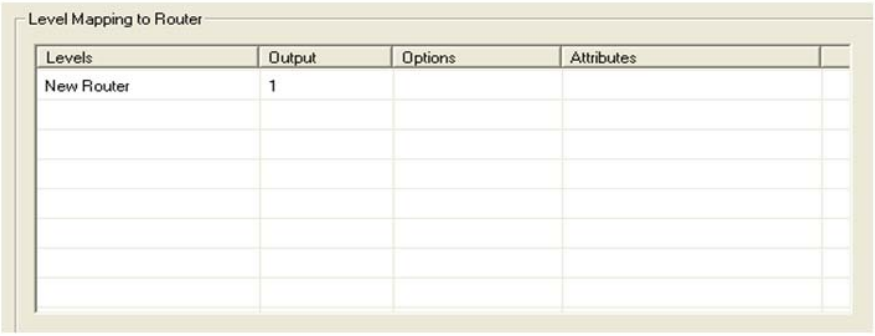


Figure 3-18.

15. Locate and launch the NFS program if it is not already running. A shortcut may be found on the desktop in a folder called 'Utah Shortcuts'. If not then it will for sure be located in Program Files in the start menu and is called ProNFS.

16. On the main system screen right click on the SC4 icon and select 'Program' and then program the device. Then click the 'Program' tab at the bottom of the screen to see the progress bar (figure 19).

Figure 3-19.

Installing the MXLator Software in the USI Folder

In order to retrieve and edit the MXLator configuration file you must first install the folder and its script files using these steps. (Skip to the next section 'Retrieving the MXLator Configuration File' when this is complete)

1. Locate and launch the Utah Install Utility which is generally found in the C:\usi directory. A shortcut may be found on the desktop in a folder called 'Utah Shortcuts' (see Figure 20). **NOTE:** If this is not on your system then refer to Chapter 1 of the 'System Installation Guide' and follow the steps to install the system software.

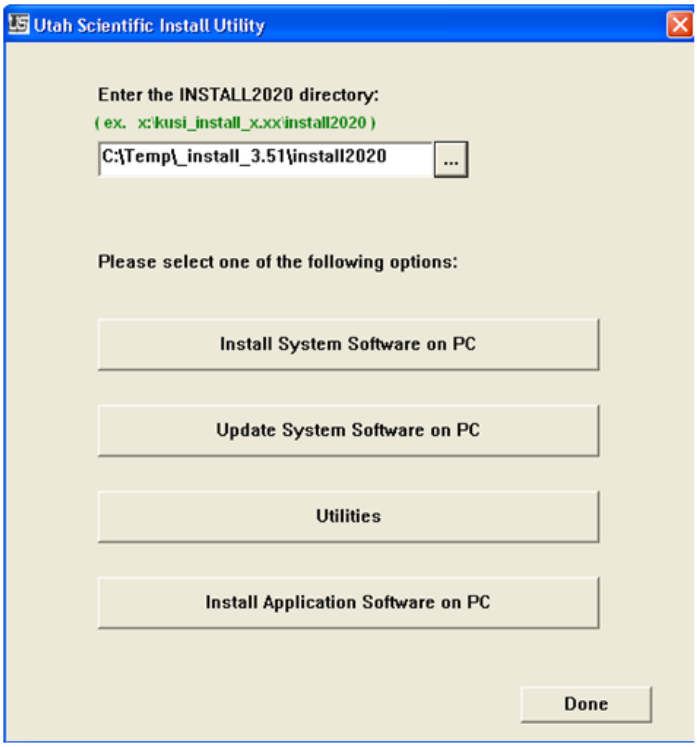


Figure 3-20.

2. Click the square button next to the directory lookup window and browse to the ‘Install2020’ folder which is located on the system CD shown in figure 21 example below.

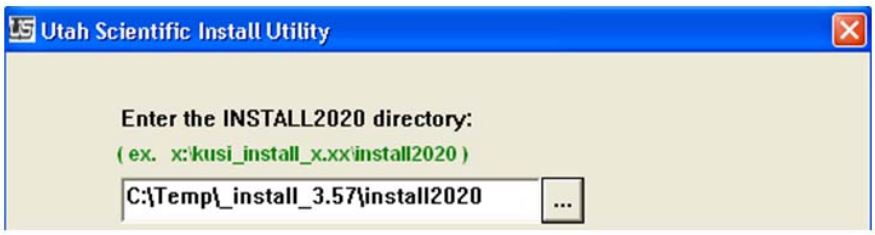


Figure 3-21.

3. Click the button labeled ‘Install System Software on PC’ (figure 22) and put the desired name in the ‘Call’ box (figure 23). Then drop down the system selection (Figure 23) list using the arrow and choose the MXLATOR item.

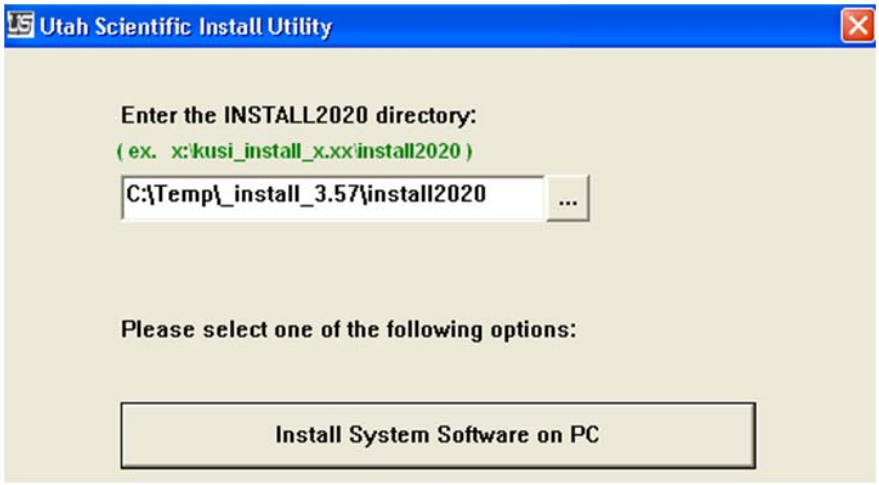


Figure 3-22.

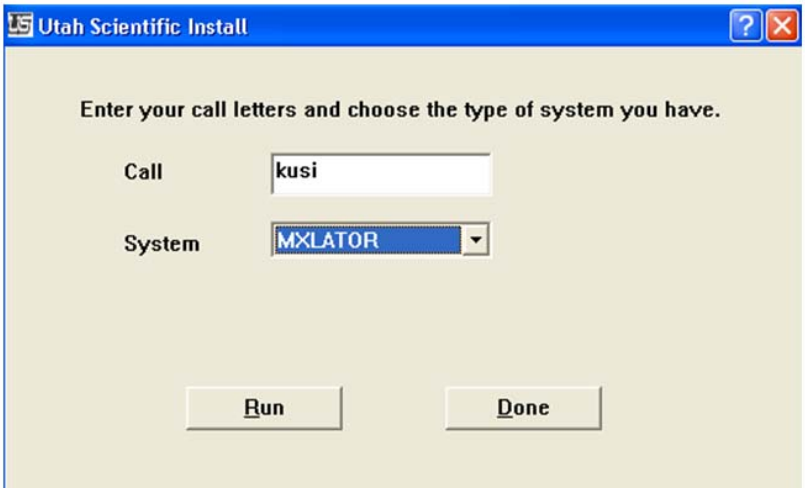


Figure 3-23.

4. Click the 'Done' button and multiple files should scroll by in a dos window. This will conclude the installation of the files needed for retrieving and configuring the MXLator.

Retrieving the MXLator Configuration File

Perform the following steps to retrieve the configuration file from the MXLator.

1. Locate and launch the NFS program. A shortcut may be found on the desktop in a folder called ‘Utah Shortcuts’. If not then it will for sure be located in Program Files in the start menu and is called ProNFS.
2. Locate and launch the Utah Install Utility which is generally found in the C:\usi directory. A shortcut may be found on the desktop in a folder called ‘Utah Shortcuts’ (see Figure 24).

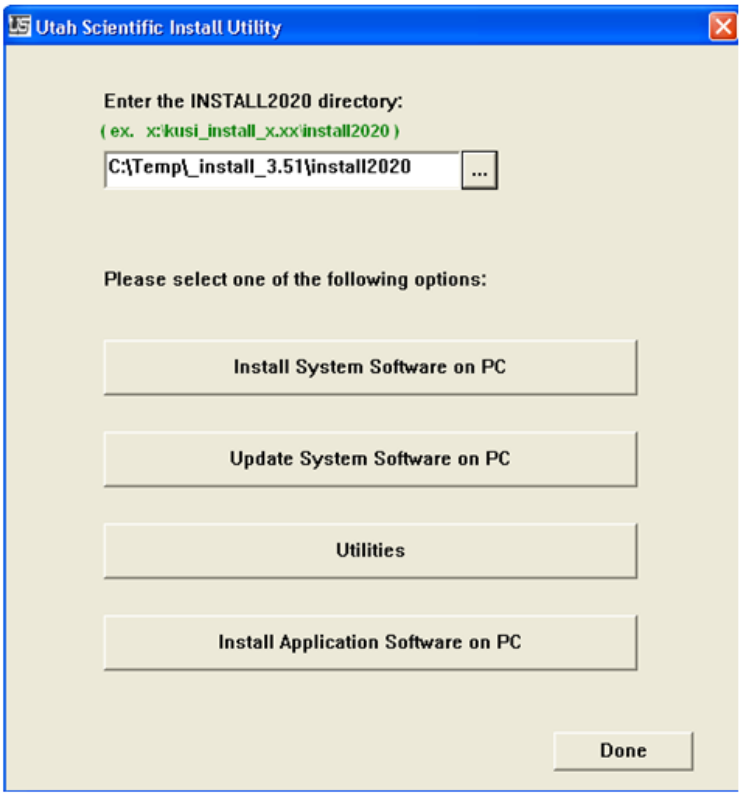


Figure 3-24.

3. Click the ‘Utilities’ button (figure 25) and then drop down the ‘Device Type Filter’ menu (figure 26) and select MXLATOR from the list.

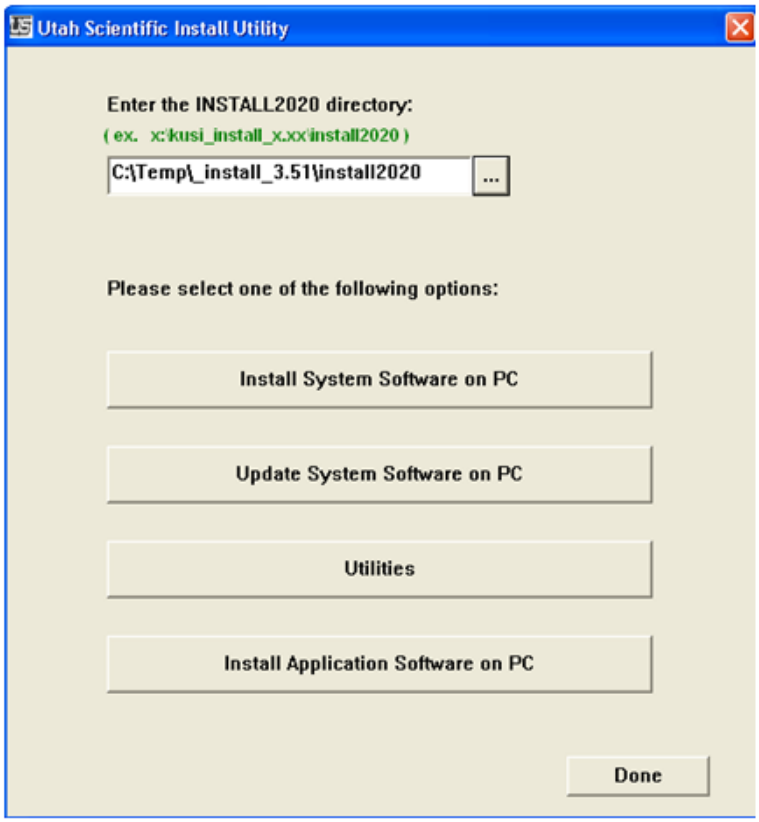


Figure 3-25.

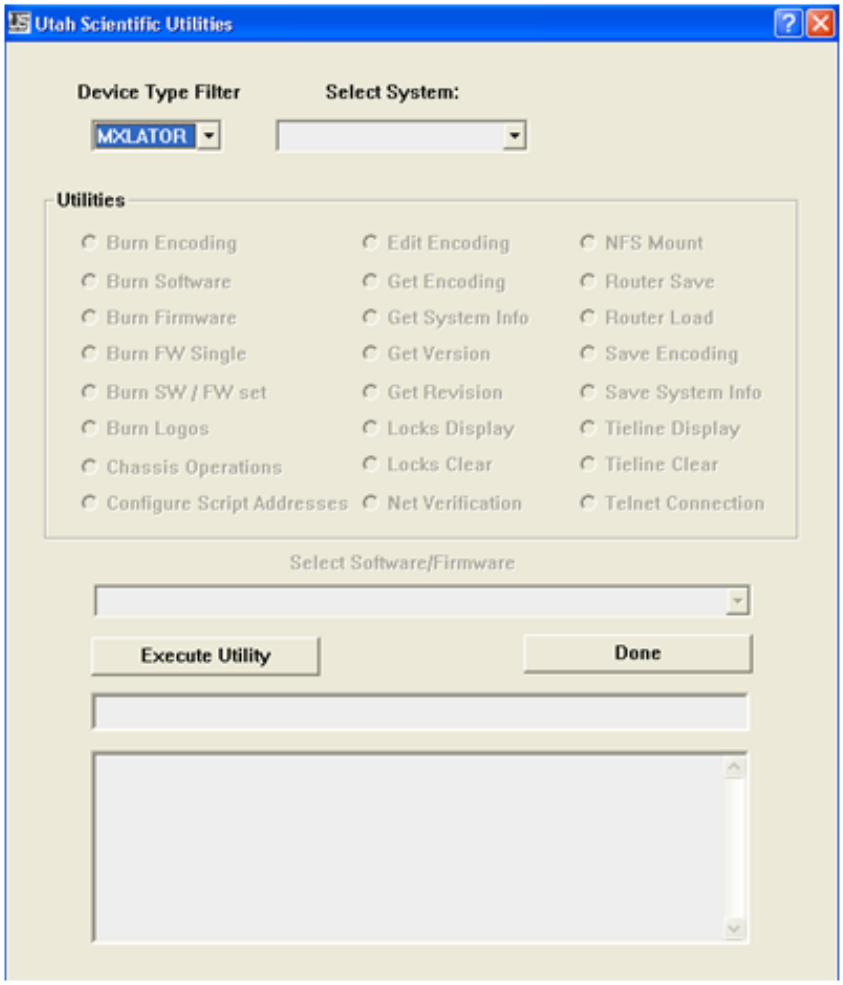

Utah Scientific Utilities
Device Type Filter
Select System:
MXLATOR
Utilities
Burn Encoding
Burn Software
Burn Firmware
Burn FW Single
Burn SW / FW set
Burn Logos
Chassis Operations
Configure Script Addresses
Edit Encoding
Get Encoding
Get System Info
Get Version
Get Revision
Locks Display
Locks Clear
Net Verification
NFS Mount
Router Save
Router Load
Save Encoding
Save System Info
Tieline Display
Tieline Clear
Telnet Connection
Select Software/Firmware
Execute Utility
Done

Figure 3-26.

4. Click on the ‘Select System’ drop down menu and then click on the file called ‘xxxx-mxlator01’ (xxxx is short for the call letters). See Figure 27 below.

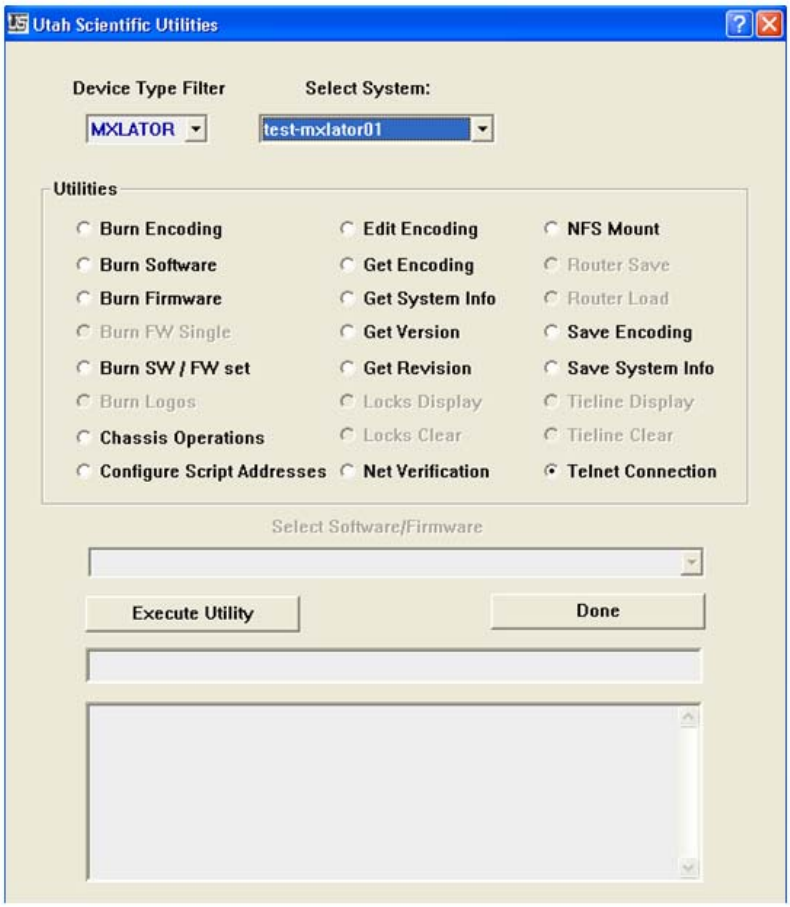


Figure 3-27.

5. **NOTE**: If there is not a system type found in step 4 then you will need to install one

6. Click on the ‘Get Encoding’ (figure 28) item on the main screen and then click the ‘Execute Utility’ button (figure 29) in the lower half of the utility screen. The retrieval should complete with a green message in the message bar below the Execute Utility button stating ‘Operation Completed Successful’.

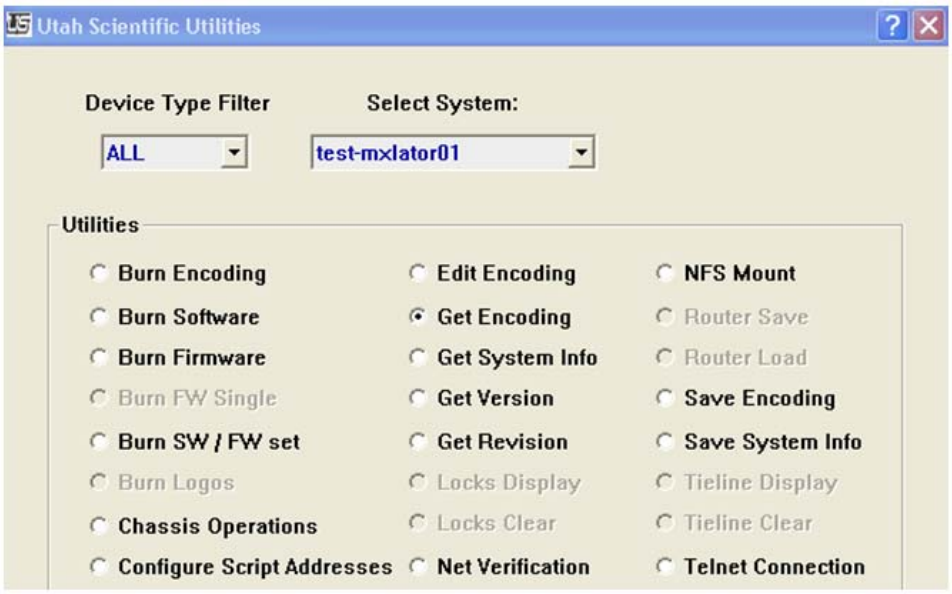


Figure 3-28.

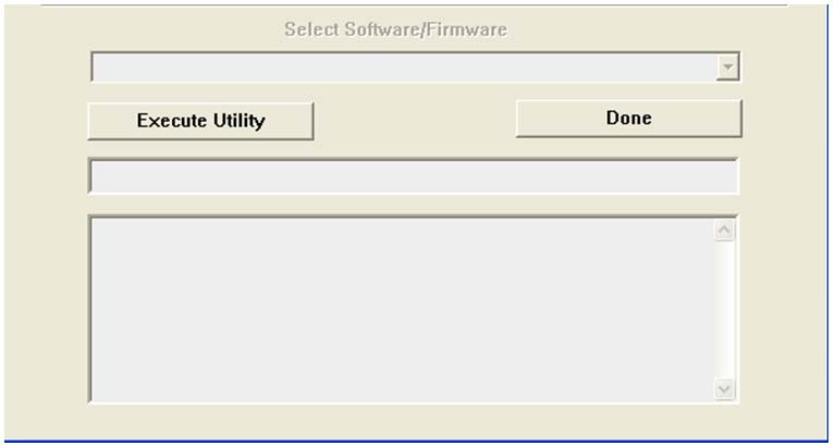


Figure 3-29.

7. Click on the 'Edit Encoding' (figure 30) item on the main screen and then click the 'Execute Utility' button (figure 31) in the lower half of the utility screen. This will open the configuration file in WordPad where all the changes will be made manually.

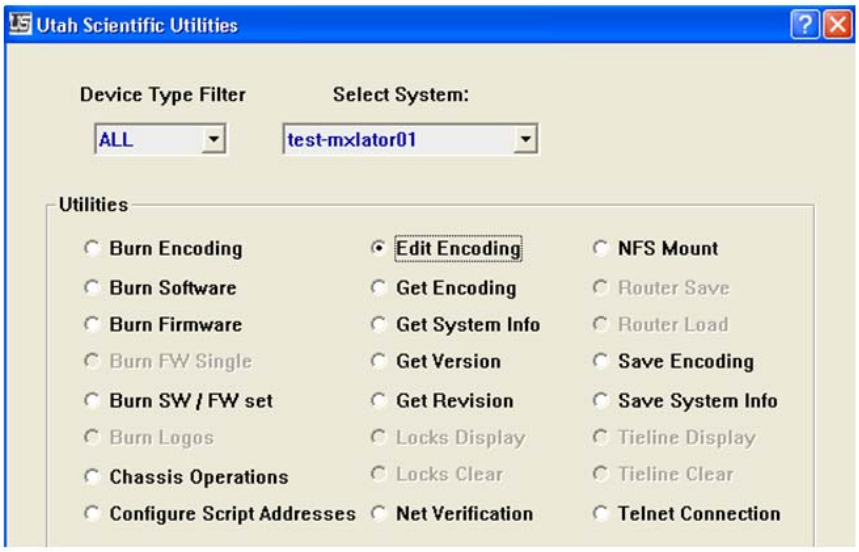


Figure 3-30.

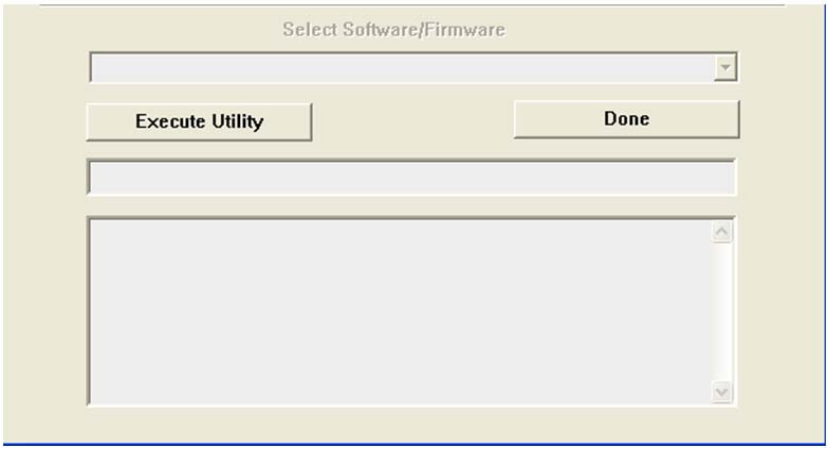


Figure 3-31.

Editing the MXLator Configuration File

Perform the following steps to edit the configuration file from the MXLator after performing step 6 above in the retrieval process.

Each section that is to be configured and used will have a header at the top of that section with [ ] brackets around it. Each section will have a brief description after the header preceded by pound # signs. These are comments and are for read only purposes. The actual usable section will follow this and will not have the pound # signs. **NOTE**: Make sure to leave the @ sign before any data that is changed.

The following tables have been extracted from the MXLator configuration file. When editing these files there will be details, examples and descriptions in parenthesis to describe important information.

MX-Bus

Note: The following section is required for most configurations. As a rule, any section that has the USI_SOURCE set to MX will require this section. This is the section that is used to convert the desired protocol into the Utah MX bus. The SC4/400 controller connects to the MX-Bus port on the MXLator unit and regardless of who is the master and who is the slave, this section defines which levels that are set up in the SC4/400 will be switched. **NOTE**: The MX bus outputs and levels are all 0 based. Create one LEVEL_MAPPING line for each level that will be used, including all interfaces below. Example: if there were one SC-BUS router and one DB-BUS router that will connect to the MXLator then there must be one LEVEL_MAPPING line created for each.

[MX_BUS_DATA_START]
ENABLED: @< TRUE | FALSE

- FUNCTIONALITY - This enables or disables the MX bus interface. Almost all routers require this to be "TRUE" (The UCI-400 and other future expansion types will require this to be FALSE). If set to FALSE the read back will still function, no takes/refreshes from the MX bus will be used.
- PARAMETERS - There is only one parameter. "TRUE" enables the bus. "FALSE" disables the bus.

READ_BACK: @0x0060

- FUNCTIONALITY - This is the upper 7 bits of the MX bus read back. It should be set to 0x60.
- PARAMETERS - There is only one parameter. "0x0060"

READ_BACK_STATUS:@< 0 | 1 | STATUS >

- FUNCTIONALITY - This is the lower bit of the MX bus read back. It has three options; hard code a '0', hard code a '1' or use the confirm 'STATUS' from a legacy router. This bit must be set to the capability of the MX bus SC4 controller depending on the router type. This should be set to '1'.
- PARAMETERS - There is only one parameter. "Use 0 or 1 for a fixed value. If you want legacy confirm, use STATUS".

DISCONNECT_INPUT:@< disconnect input number >

- FUNCTIONALITY - This is a value received from the MX bus that will be converted into a disconnect command on specific routers. Not all routers will support disconnect commands. The SC and DB bus will send out a 0x0C when this input is received on the MX bus for them. This is the input number that will be mapped into a disconnect command.
- PARAMETERS - There is only one parameter which is the number of choice for the disconnect to be switched to.

LEVEL_MAPPING: @<MX level>, <MX output start>, <MX output length>

- FUNCTIONALITY - This specifies the blocks of outputs that the MXLator will capture from the MX bus. The MXLator will capture any refresh or takes within these outputs. They will be mapped to specific routers thru the specific router configuration. It is possible to grab more data than what will be mapped to a router. Only one output block per level can be specified. This configuration can be repeated for each of the 16 MX levels. Note that only the outputs can be configured, no input validation occurs.
- PARAMETERS - There are three parameters. **"MX level"** which is the level on the MX bus set up in the SC4 router table. **"MX output start"** the first output assigned to be used as the physical router. This number is usually set to 0 but whatever the number is set to, is where the switching will begin and will also need to match in the destination tables. **"MX output length"** is set to the total size of the router outputs. Whatever this number is set to is how many outputs are able to switch.

SC-BUS (Utah AVS-2 Router)

The following section is for use with Utah Scientific AVS-2 routers and AVS-2 DATA routers. **NOTE**: All MX and SC levels and outputs are 0 relative. SC levels and outputs are limited to a total of 8 levels and 320 outputs. Create one MX_SC_MAPPING line for each physical AVS-2 level to be used and create one MX_SCDR_MAPPING line for each physical AVS-2 DATA level to be used. Create one REFRESH_LEVEL for each AVS-2 level to be used. This number must match the SC and SCDR level numbers. Create one REFRESH_DECADE line for each block of ten outputs. These decades only have the first number of that decade in the line.

[SC_BUS_DATA_START]
ENABLED: @< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the SC bus interface. Must be set to TRUE.
- PARAMETERS - There is only one parameter. "TRUE" enables the bus. "FALSE" disables the bus.

MX_SC_MAPPING: @<MX level>, <SC level>, <MX output start>, <MX output length>, <SC output offset>, <SC input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the SC will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Only one block per MX level can be specified for SC use. This line can be repeated for each of the 16 MX levels however only 8 physical SC levels may be used.
- PARAMETERS - There are six parameters. **"MX level".** This level on the MX bus will be mapped by the MXLator to a SC level and must match the level in the MX_BUS_DATA_START table. The MX bus levels are 0 relative. Each MX level can be specified only once for SC mapping. **"SC level"** is the physical AVS-2 level that will be used for the SC bus. The SC bus levels are 0 relative. **"MX output start"** is the first output on the MX bus that will map to an SC output. Outputs on the MX bus are 0 relative. Outputs on the SC bus are 0 relative. This number must match the output start in the MX_BUS_DATA_START table. **"MX output length"** is set to the total size of the router outputs. Whatever this number is set to is how many outputs are able to switch. This number must match the output start in the MX_BUS_DATA_START table. **"SC output offset" -** This number will be added to the MX output number and this will be used as the SC output. This can be a positive or negative number. **"SC input offset"**. This number will be added to the MX input number and this will be used as the SC input. This can be a positive or negative number.

MX_SCDR_MAPPING:@<MX level>, <SC level>, <MX output start>, <MX output length>,<SC output offset>, <SC input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the SC DATA ROUTER will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Only one block per MX level can be specified for SC DATA ROUTER use. This line can be repeated for each of the 16 MX levels however only 8 physical SC levels may be used.
- PARAMETERS - There are six parameters. **"MX level".** This level on the MX bus will be mapped by the MXLator to a SC level and must match the level in the MX_BUS_DATA_START table. The MX bus levels are 0 relative. Each MX level can be

specified only once for SC mapping. **"SC level"** is the physical AVS-2 DATA level that will be used for the SC bus. The SC bus levels are 0 relative. **"MX output start"** is the first output on the MX bus that will map to an SC output. Outputs on the MX bus are 0 relative. Outputs on the SC bus are 0 relative. This number must match the output start in the MX_BUS_DATA_START table. **"MX output length"** is set to the total size of the router outputs. Whatever this number is set to is how many outputs are able to switch. This number must match the output start in the MX_BUS_DATA_START table. **"SC output offset" -** This number will be added to the MX output number and this will be used as the SC output. This can be a positive or negative number. **"SC input offset"**. This number will be added to the MX input number and this will be used as the SC input. This can be a positive or negative number.

SCDR_AUTO_DISCONNECT: @ <TRUE or FALSE>

- FUNCTIONALITY - The data router can operate in two modes in handling outputs that have their inputs switched to other outputs. These output can have their source disconnected (most common use), or they can be left alone, still switched to an input that is being used for another output. Auto disconnect allows a specific port to change to a new port without manual intervention.
- PARAMETERS - There is only one parameter. "**TRUE**" will enable the auto disconnect function. "**FALSE**" will disable the auto disconnect function.

SCDR_EXCLUSIVE_LOCK: @ <TRUE or FALSE >

- FUNCTIONALITY - The data router can stop users from changing outputs that are currently in use. This lock can be enabled or disabled.
- PARAMETERS - There is only one parameter. "**TRUE**" will stop changes to outputs that currently have a source (something other than the disconnect source). "**FALSE**" allows changes to outputs that are currently in use.

REFRESH_LEVEL: @< 0 thru 7 >

- FUNCTIONALITY - The SC bus levels need to be refreshed. This specifies what levels will be refreshed on the SC bus. More than one of these lines can be specified. Each line specifying a different level. This is the physical AVS-2 router level and DATA router level. Levels are 0 based.
- PARAMETERS - There is only one parameter. The level to refresh. Valid levels are 0 thru 7. Only one level per line. Use additional lines for additional levels.

REFRESH_DECADE:@< 0 thru 31 >

- FUNCTIONALITY - The SC bus outputs need to be refreshed. This specifies what output decades will be refreshed on the SC bus. More than one of these lines can be specified, up to 32. Each line specifying a decade.
- PARAMETERS - There is only one parameter. The decade to refresh. Valid decades are 0 thru 31. Only one decade per line. Use additional lines for additional decades. 0 would be 0-9, 1 would be 10-19, etc up to 31 which would be 310-319 for a total of 320 outputs.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. "**MX**" for MX bus protocol or "**RCP**" for Ethernet type protocol also known as RCP4. MX bus will always be used for AVS-2 routers.

DB-BUS (Utah AVS-1 Router)

The following section is for use with Utah Scientific AVS-1 and AVS-1B routers. **NOTE:** All MX and DB levels and outputs are 0 relative. DB levels and outputs are limited to a total of 8 levels and 320 outputs. Create one MX_DB_MAPPING line for each physical DB level to be used. Create one REFRESH_LEVEL for each DB level to be used. This number must match the DB level numbers. Create one REFRESH_DECADE line for each block of ten outputs. These decades only have the first number of that decade in the line.

[DB_BUS_DATA_START]
ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the DB bus interface.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the bus. "**FALSE"** disables the bus.

320_OUTPUTS:@< TRUE | FALSE >

- FUNCTIONALITY - The DB bus can run as 320 x 160 x 8 or 320 x 320 x 7.
- PARAMETERS - There is only one parameter. You can enable the 320 output mode or the 160 output mode. **TRUE** will enable 320 x 320 x 7. **FALSE** will enable the 320 x 160 x 8

25_DATA_BITS:@< TRUE | FALSE >

- FUNCTIONALITY - The DB bus can run in 24 or 25 bit data bus.
- PARAMETERS - There is only one parameter. You can enable the 24 or 25 bit mode. **TRUE** will enable 25 bit mode (newer equipment such as PL320 only). **FALSE** will enable 24 bit (old equipment such as PL160 but also PL320).

MX_DB_MAPPING:@<MX level>, <DB level>, <MX output start>, <MX output length>,<DB output offset>, <DB input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the DB will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Only one block per MX level can be specified for DB use. This line can be repeated for each of the 16 MX levels.
- PARAMETERS - There are six parameters. **"MX level".** This level on the MX bus will be mapped by the MXLator to a DB level and must match the level in the MX_BUS_DATA_START table. The MX bus levels are 0 relative. Each MX level can be specified only once for DB mapping. **"DB level"** is the physical AVS-1 level that will be used for the DB bus. The SC bus levels are 0 relative. **"MX output start"** is the first output on the MX bus that will map to a DB output. Outputs on the MX bus are 0 relative. Outputs on the DB bus are 0 relative. This number must match the output start in the MX_BUS_DATA_START table. **"MX output length"** is set to the total size of the router outputs. Whatever this number is set to is how many outputs are able to switch. This number must match the output start in the MX_BUS_DATA_START table. **"DB output offset".** This number will be added to the MX output number and this will be used as the DB output. This can be a positive or negative number. **"DB input offset"**. This number will be added to the MX input number and this will be used as the SC input. This can be a positive or negative number.

REFRESH_LEVEL:@< 0 thru 7 >

- FUNCTIONALITY - The DB bus needs to be refreshed. This specifies what levels will be refreshed on the DB bus. More than one of these lines can be specified. Each line specifying a different level. This is the physical AVS-1 router level. Levels are 0 based.
- PARAMETERS - There is only one parameter. The level to refresh. Valid levels are 0 thru 7. Only one level per line. Use additional lines for additional levels.

REFRESH_DECADE:@< 0 thru 31 >

- FUNCTIONALITY - The DB bus needs to be refreshed. This specifies what output decades will be refreshed on the DB bus. More than one of these lines can be specified, up to 32. Each line specifying a decade.
- PARAMETERS - There is only one parameter. The decade to refresh. Valid decades are 0 thru 31. Only one decade per line. Use additional lines for additional decades. 0 would be 0-9, 1 would be 10-19, etc up to 31 which would be 310-319 for a total of 320 outputs.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. "**MX**" for MX bus protocol or "**RCP**" for Ethernet type protocol also known as RCP4. MX bus will always be used for AVS-1/1B routers.

ES-BUS (Various Router Types such as Venus VM/SI3000)

The following section is for use with various router types. BC stands for 'Bridge Control'. Bridge control allows both the USI and the Third Party routers to be in control, switching either router from its control panels. NOTE: See the notes for this router type at the bottom of this document in the 'Example Config File' section.

[ESBC_BUS_DATA_START]

This section can be repeated until all 6 serial ports are allocated. Other protocols can allocate serial ports included in the max of 6.

ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the ESBC bus interface. For this interface to work, you must also enable the RCP3 interface.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the bus. "**FALSE**" disables the bus.

SERIAL_PORT: @<port (1 thru 6)>, <baud rate>, <bit length>, <stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The ES bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Note that the normal setting for ES bus is 38400, 8 data, 1 stop, Odd parity. Note that this port is specified by itself if the physical router is on the non-USI controller. If the physical router is on the USI controller then both this port and the SERIAL_PORT_TR must be specified (on a different physical serial port).
- PARAMETERS - There are six parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This is set to 7 or 8. **"Stop bits"**. This is set to 1 or 2. **"Parity"**. This is set to O for odd or E for even. **"Hardware handshake"**. This is N for none.

SERIAL_PORT_TR: @<port (1 thru 6)>, <baud rate>, <bit length>,<stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The ES bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Note that the normal setting for ES bus is 38400, 8 data, 1 stop, Odd parity. Note that this port cannot be specified by itself. When this port is specified, it is in conjunction with SERIAL_PORT above. It will be specified when the physical router is on the USI controller. The SERIAL_PORT and SERIAL_PORT_TR are specified on separate ports.
- PARAMETERS - There are six parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This is set to 7 or 8. **"Stop bits"**. This is set to 1 or 2. **"Parity"**. This is set to O for odd or E for even. **"Hardware handshake"**. This is N for none.

MX_ESBC_MAPPING: @<MX level>, <ESBC level>,<MX output start>, <MX output length>,<ESBC output offset>, <ESBC input offset>, <ESBC non-confirm mx source>

- FUNCTIONALITY - This specifies the blocks of outputs that the ESBC will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Up to 16 of these may be specified. The USI level can be repeated, ES level cannot be repeated. The ESBC mapping is associated with the SERIAL_PORT protocol. These physical cross points are on the ES controller, not USI controller. The ESBC level is 0 based and is found in the Jupiter controller setup software in the 'Level Set' that is created for the serial router type. The 'Level Set' will have row numbers next to each router type. This is the level number minus 1. So if the row number is 1 then the number in the ESBC Level would be a 0.
- PARAMETERS - There are seven parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to an ESBC level. The MX bus levels are 0 relative. The MX level can be specified more than once. **"ESBC level"**. This is the level that will be used for the ESBC bus. The ESBC bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to an ESBC output. Outputs on the MX bus are 0 relative. Outputs on the ESBC bus are 0 relative. **"MX output length"**. This is the output block size. **"ESBC output offset"**. This number will be added to the MX output number and this will be used as the ESBC output. This can be a positive or negative number. **"ESBC input offset"**. This number will be added to the MX input number and this will be used as the ESBC input. This can be a positive or negative number. **"ESBC non-confirm source"**. If the MXLator cannot switch the ES router, the MXLator will change the USI side status of all of the outputs associated with this ES router to the input specified here. It will not change the router to this source. If no input is specified here then no status change will occur on the USI router status when unable to make ES router changes.

MX_ESBC_LCL_MAPPING: @<MX level>, <ESBC_LCL level>, <MX output start>,<MX output length>, <ESBC_LCL output offset>, <ESBC_LCL input offset>, <ESBC_LCL non-confirm mx source>

- FUNCTIONALITY - This specifies the blocks of outputs that the ESBC_LCL will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Up to 16 of these may be specified. The USI level can be repeated, ES level cannot be repeated. The ESBC_LCL mapping is associated with the SERIAL_PORT_TR protocol. These physical cross points are on the USI controller.

- PARAMETERS - There are seven parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a ESBC_LCL level. The MX bus levels are 0 relative. The MX level can be specified more than once. **"ESBC_LCL level"**. This is the level that will be used for the ESBC_LCL bus. The ESBC_LCL bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to an ESBC_LCL output. Outputs on the MX bus are 0 relative. Outputs on the ESBC_LCL bus are 0 relative. **"MX output length"**. This is the output block size. **"ESBC_LCL output offset"**. This number will be added to the MX output number and this will be used as the ESBC_LCL output. This can be a positive or negative number. **"ESBC_LCL input offset"**. This number will be added to the MX input number and this will be used as the ESBC_LCL input. This can be a positive or negative number. **"ESBC_LCL non-confirm source"**. If the MXLator cannot switch the USI router. The MXLator will change the ES side status of all of the outputs associated with this ES router to the input specified here. It will not change the router to this source. If no input is specified here then no status change will occur on the ES router status when unable to make USI router changes.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS-- There is only one parameter. The source of the SC4 takes. **MX** or **RCP**

ES-BUS (Various Router Types such as the Venus VM/SI3000)

The following section is for use with various router types. CM stands for 'Control Master'. It is used in Master mode where the SC4 controls the Venus routers. This protocol has a continuous refresh of all the outputs that are in the range. If only some outputs are needed then the mapping table should be set to just that size and then in the router level only the outputs that will need to be switched should be in that level. NOTE: See the notes for this

router type at the bottom of this document in the 'Example Config File' section.

[ESBCM_BUS_DATA_START]

This section can be repeated until all 6 serial ports are allocated. Other protocols can allocate serial ports included in the max of 6.

ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the ESBCM (ESBC Master) bus interface. For this interface to work, you must also enable the RCP3 interface.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the bus. "**FALSE**" disables the bus.

SERIAL_PORT: @<port (1 thru 6)>, <baud rate>, <bit length>,<stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The ES bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Note that the normal setting for ES bus is 38400, 8 data, 1 stop, Odd parity. Note that this port is specified by it-self if the physical router is on the non-USI controller. If the physical router is on the USI controller then both this port and the SERIAL_PORT_TR must be specified (on a different physical serial port).
- PARAMETERS - There are six parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This can be 7 or 8. **"Stop bits"**. This can be 1 or 2. **"Parity"**. This can be O for odd, E for even, N for none. **"Hardware handshake"**. This is N for none or H for rtscts hardware handshake.

SERIAL_PORT_TR: @<port (1 thru 6)>, <baud rate>, <bit length>,<stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The ES bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Note that the normal setting for ES bus is 38400, 8 data, 1 stop, Odd parity. Note that this port cannot be specified by itself. When this port is specified, it is in conjunction with SERIAL_PORT above. It will be specified when the physical router is on the USI controller. The SERIAL_PORT and SERIAL_PORT_TR are specified on separate ports.

- PARAMETERS - There are six parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This can be 7 or 8. **"Stop bits"**. This can be 1 or 2. **"Parity"**. This can be O for odd, E for even, N for none. **"Hardware handshake"**. This is N for none or H for rtscts hardware handshake.

MX_ESBCM_MAPPING: @<MX level>, <ESBCM level>,<MX output start>, <MX output length>,<ESBCM output offset>, <ESBCM input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the ESBCM will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Up to 16 of these may be specified. The USI level can be repeated, ES level cannot be repeated. The ESBCM mapping is associated with the SERIAL_PORT protocol. These physical cross points are on the ES controller, not USI controller. The ESBC level is 0 based and is found in the Jupiter controller setup software in the 'Level Set' that is created for the serial router type. The 'Level Set' will have row numbers next to each router type. This is the level number minus 1. So if the row number is 1 then the number in the ESBC Level would be a 0.
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a ESBCM level. The MX bus levels are 0 relative. The MX level can be specified more than once. **"ESBCM level"**. This is the level that will be used for the ESBCM bus. The ESBCM bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to an ESBCM output. Outputs on the MX bus are 0 relative. Outputs on the ESBCM bus are 0 relative. **"MX output length"**. This is the output block size. **"ESBCM output offset"**. This number will be added to the MX output number and this will be used as the ESBCM output. This can be a positive or negative number. **"ESBCM input offset"**. This number will be added to the MX input number and this will be used as the ESBCM input. This can be a positive or negative number.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

ES-BUS (Thompson GVG Router)

The following section is for use with a Thompson GVG router. TRC stands for 'Tributary Confirm'. It is used in Slave mode where the GVG controls the Utah routers.

[ESTRC_BUS_DATA_START]

This section can be repeated until all 6 serial ports are allocated. Other protocols can allocate serial ports included in the max of 6.

ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the ESTRC bus interface. This interface will rarely be used. It can be used for systems that need a confirm for some cross points but not confirm all. The VM3000 controller is one of these. The [ESBC_BUS_DATA_START] ESBC_LCL_MAPPING and [ESBC_BUS_DATA_START] SERIAL_PORT_TR will duplicate this function with added status capability.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the bus. "**FALSE**" disables the bus.

SERIAL_PORT: @<port (1 thru 6)>, <baud rate>, <bit length>,<stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The ES bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Note that the normal setting for ES bus is 38400, 8 data, 1 stop, Odd parity.

- PARAMETERS - There are six parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This can be 7 or 8. **"Stop bits"**. This can be 1 or 2. **"Parity"**. This can be O for odd, E for even, N for none. **"Hardware handshake"**. This is N for none or H for rtscts hardware handshake.

MX_ESTR_MAPPING: @<MX level>, <ESTR level>, <MX output start>,<MX output length>, <ESTR output offset>, <ESTR input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the ESTR will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Up to 16 of these may be specified. The USI level can be repeated, ES level cannot be repeated. These physical cross points are on the USI controller.
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to an ESTR level. The MX bus levels are 0 relative. The MX level can be specified more than once. **"ESBTR level"**. This is the level that will be used for the ESBTR bus. The ESTR bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to an ESTR output. Outputs on the MX bus are 0 relative. Outputs on the ESTR bus are 0 relative. **"MX output length"**. This is the output block size. **"ESTR output offset"**. This number will be added to the MX output number and this will be used as the ESTR output. This can be a positive or negative number. **"ESTR input offset"**. This number will be added to the MX input number and this will be used as the ESTR input. This can be a positive or negative number.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

ES-BUS (Thompson GVG Router)

The following section is for use with a Thompson GVG router. TR stands for Tributary. It is used in Slave mode where the GVG controls the Utah routers.

[ESTR_BUS_DATA_START]
This section can be repeated until all 6 serial ports are allocated. Other
Protocols can allocate serial ports included in the max of 6.

ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the ESTR bus interface. This is the ES tributary interface. This allows non-USI controllers to switch USI equipment.
- PARAMETERS - There is only one parameter. "TRUE" enables the bus. "FALSE disables the bus.

SERIAL_PORT: @<port (1 thru 6)>, <baud rate>, <bit length>,<stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The ES bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Note that the normal setting for ES bus is 38400, 8 data, 1 stop, Odd parity.
- PARAMETERS - There are six parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This can be 7 or 8. **"Stop bits"**. This can be 1 or 2. **"Parity"**. This can be O for odd, E for even, N for none. **"Hardware handshake"**. This is N for none or H for rtscts hardware handshake.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

RCP-3 (Utah Protocol Section)

This section is for use with any system that controls the USI. It is also used with any UCI-400. Also as a rule, anytime the USI_SOURCE section is set to RCP requires this section to be used. This would be whenever the SC4 would connect and communicate to the MXLator via Ethernet and not MX Bus. Usually the only part of this that needs to be configured is the IP address which is the USI SC4 controller.

[RCP3_BUS_DATA_START]
ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the RCP3 bus interface. This must be enabled for bridge type router interfaces to work. This includes ES, Prosan and SC Data Router buses.
- PARAMETERS - There is only one parameter. "TRUE" enables the bus. "FALSE disables the bus.

NUMBER_FORMAT:@< IP xxx.xxx.xxx.xxx >
FUNCTIONALITY - The RCP3 is a protocol that runs over Ethernet. This is the IP of the USI controller (SC4).
PARAMETERS - There is only one parameter. The IP of the USI control (SC4).

REFRESH_TIME:@< milliseconds >

- FUNCTIONALITY - This is the interval between requesting status from the SC4. It is in mili seconds. This is the refresh for RCP sources.
- PARAMETERS - There is only one parameter. The time between requests.

REFRESH_BLOCK:@< block size>

- FUNCTIONALITY - This is the size of the refresh block from the SC4.
- PARAMETERS - There is only one parameter. The block size. (Default is set to 3 and should be sufficient)

PROSAN (Sandar Router)

The following section is for use with a Sandar router. It is used in Slave mode where the Sandar controls the Utah routers.

[PROSAN_BUS_DATA_START]

This section can be repeated until all 6 serial ports are allocated. Other protocols can allocate serial ports included in the max of 6. Currently the Prosan is only capable of controlling Utah routers.

ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the PROSAN bus interface.
- PARAMETERS -- There is only one parameter. "**TRUE**" enables the bus. "**FALSE**" disables the bus.

SERIAL_PORT: @<port (1 thru 6)>, <baud rate>, <bit length>, <stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The Prosan bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Note that the normal setting for Prosan bus is 9600, 7 data, 1 stop, Odd parity.
- PARAMETERS - There are six parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"** is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This can be 7 or 8. **"Stop bits"**. This can be 1 or 2. **"Parity"** is O for odd, E for even, N for none. **"Hardware handshake"** is N for none or H for rtscts hardware handshake.

NUMBER_FORMAT:@< 2 or 3 >

- FUNCTIONALITY - The Prosan protocol can be configured with numbers that are under 100 or numbers that are under 1000. This is a configuration specified by the Prosan device.
- PARAMETERS - There is only one parameter. The number of digits in the numbers. 2 for Prosan devices that use numbers under 100. 3 for Prosan devices that use numbers under 1000.

MX_PROSAN_MAPPING: @<MX level>, < PROSAN level>, <MX output start>, <MX output length>, < PROSAN output offset>, < PROSAN input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the PROSAN will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Only one block per MX level can be specified for PROSAN use. This line can be repeated for each of the 16 MX levels.
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a PROSAN level. The MX bus levels are 0 relative. Each MX level can be specified only once for PROSAN mapping. **"PROSAN level"**. This is the level that will be used for the PROSAN bus. The PROSAN bus levels are 'A' relative. Valid Prosan levels are 'A' thru 'H' inclusive. **"MX output start"**. This is the first output on the MX bus that will map to a PROSAN output. Outputs on the MX bus are 0 relative. **"MX output length"**. This is the output block size. **"PROSAN output offset"**. This number will be added to the MX output number and this will be used as the PROSAN output. This can be a positive or negative number. Outputs on the PROSAN bus are 1 relative. This mapping must make the 0 to 1 relative map. This will typically be an odd number. **"PROSAN input offset"**. This number will be added to the MX input number and this will be used as the PROSAN input. This can be a positive or negative number. Inputs on the PROSAN bus are 1 relative. This mapping must make the 0 to 1 relative map. This will typically be an odd number.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

SIERRA-BUS (Sierra Router)

The following section is for use with a Sierra router. It is used in Master mode where the SC4 controls the Sierra routers. **NOTE:** The sierra switches slow. The Sierra manual says that status make take seconds.

[SIERRA_BUS_DATA_START]
ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the SIERRA bus operation.
- PARAMETERS - There is only one parameter. "TRUE" enables the operation. "FALSE disables the operation.

SERIAL_PORT: @<port (1 thru 6)>, <baud rate>, <bit length>,<stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The SIERRA bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Note that the normal setting for SIERRA bus is 9600, 8 data, 1 stop, Odd parity.
- PARAMETERS - There are six parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This can be 7 or 8. **"Stop bits"**. This can be 1 or 2. **"Parity"**. This can be O for odd, E for even, N for none. **"Hardware handshake"**. This is N for none or H for rtscts hardware handshake.

COMMAND_START_CHARS:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the one or two character message starts/stops.
- PARAMETERS - There is only one parameter. **TRUE** will send 2 characters. **FALSE** will send one character.

MX_SIERRA_MAPPING:@<MX level>, < SIERRA level>, <MX output start>,<MX output length>, < SIERRA output offset>, < SIERRA input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the SIERRA will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Only one block per MX level can be specified for SIERRA use. This line can be repeated for each of the 16 MX levels. Note that the SIERRA is 1 relative and the MX is 0 relative.
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a SIERRA level. The MX bus levels are 0 relative. Each MX level can be specified only once for SIERRA mapping. **"SIERRA level"**. This is the level that will be used for the SIERRA bus. The SIERRA bus levels are '1' relative. **"MX output start"**. This is the first output on the MX bus that will map to an SIERRA output. Outputs on the MX bus are 0 relative. **"MX output length"**. This is the output block size. **"SIERRA output offset"**. This number will be added to the MX output number and this will be used as the SIERRA output. This can be a positive or negative number. Outputs on the SIERRA bus are 1 relative. This mapping must make the 0 to 1 relative map. This will typically be an odd number. **"SIERRA input offset"**. This number will be added to the MX input number and this will be used as the SIERRA input. This can be a positive or negative number. Inputs on the SIERRA bus are 1 relative. This mapping must make the 0 to 1 relative map. This will typically be an odd number.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.

- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

PROBEL-BUS (Probel Router)

The following section is for use with a Probel router. NS stands for network slave. It is used in Slave mode where the SC4 is controlled by the Probel routers. This interface is currently used for controlling the Utah routers from a HARMONIC NMX system and may not work with other systems.

[PROBELNS_BUS_DATA_START]
ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the Probel network slave bus operation.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the operation. "**FALSE**" disables the operation.

IP_PORT_PROTOCOL:@< port number >

- FUNCTIONALITY - This sets the port number for the TCP/IP commands. The Probel uses this port as level 0 then goes up one for each level. This was written for HARMONIC NMX so only level 0 works.
- PARAMETERS - There is only one parameter. The port number (**NMX uses only 2000**).

IP_PORT_ERRORS:@< port number >

- FUNCTIONALITY-- This sets the port number for the UDP errors. The Probel uses this port to send out error (SNMP type?). This was written for HARMONIC NMX so it sends nothing.
- PARAMETERS - There is only one parameter. The port number (**NMX uses only 2002**).

DEFAULT_SOURCE:@< source number >

- FUNCTIONALITY - This sets the default source. The Probel will always have a source (0 is default). USI may not have a source. This value is used in these cases.

- PARAMETERS - There is only one parameter. The default source.

MX_PROBEL_MAPPING: @<MX level>, < Probel level>, <MX output start>,<MX output length>, < Probel output offset>, < Probel input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the Probel will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Only one block per MX level can be specified for Probel use. This line can be repeated for each of the 16 MX levels. Note that the Probel is 0 relative and the MX is 0 relative.
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a Probel level. The MX bus levels are 0 relative. Each MX level can be specified only once for Probel mapping. **"PROBEL level"**. This is the level that will be used for the Probel bus. The Probel bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to an Probel output. Outputs on the MX bus are 0 relative. **"MX output length"**. This is the output block size. **"PROBEL output offset"**. This number will be added to the MX output number and this will be used as the PROBEL output. This can be a positive or negative number. Outputs on the SIERRA bus are 0 relative. **"PROBEL input offset"**. This number will be added to the MX input number and this will be used as the PROBEL input. This can be a positive or negative number.

PROBEL-BUS (Probel Router)

The following router is for use with the Probel router. SM stands for serial master. It is used in Master mode where the SC4 controls the Probel routers. NOTE: The baud rate and parity must always be 38400, 8E1, N for no flow control.

[PROBELSM_BUS_DATA_START]
ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the Probel serial master bus operation.
- PARAMETERS - There is only one parameter. "TRUE" enables the operation. "FALSE disables the operation.

SERIAL_PORT: @<port (1 thru 6)>, <baud rate>, <bit length>,<stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The Probel serial bus runs on a serial port which connects to the MXLator. This specifies which serial port it will run on the MXLator. The serial port operating parameters are specified here.
- PARAMETERS - There are five parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rate 38400 only. **"Bit length"**. This is 8 only. Stop bits. This is 1 only. **"Parity"**. This is E for even only. There may be a slight chance that the parity is set to O for odd but in most every case it will be even. **"Hardware handshake"**. This is N for none only.

MX_PROBEL_MAPPING: @<MX level>, <MX output start>,<MX output length>, < Probel output offset>, < Probel input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the Probel will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. The MX level can be used more than once, each with a different output block. Note that the Probel is only one level per serial port. The MX and Probel is 0 relative.
- PARAMETERS - There are five parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a Probel level. The MX bus levels are 0 relative. Each MX level can be specified only once for Probel mapping. **"MX output start"**. This is the first output on the MX bus that will map to an Probel output. Outputs on the MX bus are 0 relative. **"MX output length"**. This is the output block size. **"PROBEL output offset"**. This number will be added to the MX output number and this will be used as the PROBEL output. This can be a positive or negative number. Outputs on the SIERRA bus are 0 relative. **"PROBEL input offset"**. This number will be added to the MX input number and this will be used as the PROBEL input. This can be a positive or negative number.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

DATATEK-BUS (Datatek Router)

The following section is for use with the Datatek router. BC stands for Bridge Control. Bridge control allows both the USI and the GVG routers to be in control, switching either router from its control panels. This section can be repeated until all 6 serial ports are allocated. Other protocols can allocate serial ports included in the max of 6.

[DATATEKBC_BUS_DATA_START]
ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the Datatek bridge bus interface. For this interface to work, you must also enable the RCP3 interface and MX bus interface.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the bus. "**FALSE**" disables the bus.

SERIAL_PORT: @<port (1 thru 6)>, <baud rate>, <bit length>,<stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The Datatek bridge bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Note that the normal setting for Datatek bus is RS232, 9600 baud, 8 data, 1 stop, no parity, no hardware flow. One port is set to 38400.
- PARAMETERS - There are six parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This can be 7 or 8. **"Stop bits"**. This can be 1 or 2. **"Parity"**. This can be O for odd, E for even, N for none. **"Hardware handshake"**. This is N for none or H for rtscts hardware handshake.

MX_DATATEK_MAPPING: @<MX level>, <Datatek level>,<MX output start>, <MX output length>,<Datatek output offset>, <Datatek input offset>, <Datatek non-confirm mx source>

- FUNCTIONALITY - This specifies the blocks of outputs that the Datatek will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Up to 16 of these may be specified. The USI level can be repeated, Datatek level cannot be repeated. These physical cross points are on the Datatek controller, not USI controller.
- PARAMETERS - There are seven parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a ESBC level. The MX bus levels are 0 relative. The MX level can be specified more than once. **"Datatek level"**. This is the level that will be used for the Datatek bus. The Datatek bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to an Datatek output. Outputs on the MX bus are 0 relative. Outputs on the Datatek bus are 0 relative. **"MX output length"**. This is the output block size. **"Datatek output offset"**. This number will be added to the MX output number and this will be used as the Datatek output. This can be a positive or negative number. **"Datatek input offset"**. This number will be added to the MX input number and this will be used as the Datatek input. This can be a positive or negative number. **"Datatek non-confirm source"**. If the MXLator cannot switch the Datatek router, the MXLator will change the USI side status of all of the outputs associated with this Datatek physical router to the input specified here. It will not change the router to this source. If no input is specified here then no status change will occur on the USI router status when unable to make Datatek router changes. This is an optional parameter.

MX_DATATEK_LCL_MAPPING: @<MX level>, <Datatek level>, <MX output start>,<MX output length>, <Datatek output offset>, <Datatek input offset>, <Datatek non-confirm mx source>

- FUNCTIONALITY - This specifies the blocks of outputs that the Datatek will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Up to 16 of these may be specified. The USI level can be repeated, Datatek level cannot be repeated. These physical cross points are on the USI controller.
- PARAMETERS - There are seven parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a Datatek level. The MX bus levels are 0 relative. The MX level can be specified more than once. **"Datatek level"**. This is the level that will be used for the Datatek bus. The Datatek bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to an Datatek output. Outputs on the MX bus are 0 relative. Outputs on the Datatek bus are 0 relative. **"MX output length"**. This is the output block size. **"Datatek output offset"**. This number will be added to the MX output number and this will be used as the Datatek output. This can be a positive or negative number. **"Datatek input offset"**. This number will be added to the MX input number and this will be used as the Datatek input. This can be a positive or negative number. **"Datatek non-confirm source"**. If the MXLator cannot switch the USI router. The MXLator will change the Datatek side status of all of the outputs associated with this USI router to the input specified here. It will not change the router to this source. If no input is specified here then no status change will occur on the Datatek router status when unable to make USI router changes. This is an optional parameter

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

GV-BUS (Thompson GV Router)

The following section is for use with the Thompson GVG router. SN stands for Serial Network. It is used in Slave mode where the GVG controls the Utah routers on a network. USI_SOURCE must be set to RCP.

[GVNSN_BUS_DATA_START]

ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the GV native protocol network slave bus operation.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the operation. "**FALSE**" disables the operation.

PORT: @< Port >

- FUNCTIONALITY - This sets the IP port to listen on for connections. Default is 12345
- PARAMETERS - There is only one parameter. The IP port to listen on

LEVEL_4_ACK:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the LEVEL 4 ACK in the protocol. There are also commands in the protocol to override this setting. To make the GVG test GUI work, this needs to be set to TRUE.
- PARAMETERS - There is only one parameter. Use **TRUE** to configure to send Level 4 acks, Use **FALSE** to configure to not send Level 4 acks.

TIME_OUT:@< number of seconds >

- FUNCTIONALITY - This sets the protocol timeout. This is the number of seconds with no messages before the connection will be broken. Set this to 0 to turn off this function.
- PARAMETERS - There is only one parameter. The number of seconds for this timer.

MX_GVNMN_MAPPING:@<MX level>, < GV level>, <MX output start>,<MX output length>, < GV output offset>, < GV input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the GV will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Note that the GV is 0 relative and the MX is 0 relative.
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a Probel level. The MX bus levels are 0 relative. **"GV level"**. This is the level that will be used for the GV bus. The GV bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to an GV output. Outputs on the MX bus are 0 relative. **"MX output length"**. This is the output block size. **"GV output offset"**. This number will be added to the MX output number and this will be used as the GV output. This can be a positive or negative number. Outputs on the GV bus are 0 relative. **"GV input offset"**. This number will be added to the MX input number and this will be used as the GV input. This can be a positive or negative number. GV inputs are 0 relative.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS-- There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

GV-BUS (Thompson GV Router)

The following section is for use with the Thompson GVG router. MN stands for Master Network. It is used in Master mode where the SC4 controls the Thompson routers over network.

[GVNMN_BUS_DATA_START]
ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the GV native network Master bus operation.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the operation. "**FALSE**" disables the operation.

IP_ONE:@< IP >

- FUNCTIONALITY - This sets the IP for the primary controller.
- PARAMETERS - There is only one parameter. The IP of the primary controller

IP_TWO:@< IP >

- FUNCTIONALITY - This sets the IP for the backup or secondary controller.
- PARAMETERS - There is only one parameter. The IP of the backup or secondary controller

MX_GVNMN_MAPPING: @<MX level>, < GV level>, <MX output start>,<MX output length>, < GV output offset>, < GV input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the GV will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Note that the GV is 0 relative and the MX is 0 relative.
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a Probel level. The MX bus levels are 0 relative. **"GV level"**. This is the level that will be used for the GV bus. The GV bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to an GV output. Outputs on the MX bus are 0 relative. **"MX output length"**. This is the output block size. **"GV output offset"**. This number will be added to the MX output number and this will be used as the GV output. This can be a positive or negative number. Outputs on the GV bus are 0 relative. **"GV input offset"**. This number will be added to the MX input number and this will be used as the GV input. This can be a positive or negative number. GV inputs are 0 relative.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.

- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

PESA-BUS (PESA Router)

The following section is for use with the PESA router. SM stands for serial master. It is used in Master mode where the SC4 controls the PESA routers serially.

[PESASM_BUS_DATA_START]
ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the Pesa serial master bus operation.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the operation. "**FALSE**" disables the operation.

SERIAL_PORT: @<port (1 thru 6)>, <baud rate>, <bit length>,<stop bits>, <parity>, <hardware handshake>

- FUNCTIONALITY - The Pesa serial bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Pesa serial default RS232,9600,8,2,N,H
- PARAMETERS - There are five parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This can be 7 or 8. **"Stop bits"**. This can be 1 or 2. **"Parity"**. This can be O for odd, E for even, N for none. **"Hardware handshake"**. This is N for none or H for rtscts hardware handshake.

MX_GVNMN_MAPPING: @<MX level>, < Pesa level>, <MX output start>,<MX output length>, < Pesa output offset>,
< Pesa input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the Pesa will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Note that the Pesa is 1 relative and the MX is 0 relative for inputs and outputs. The levels should be mapped as 0 relative for both, (the physical Pesa router may be labeled with levels that are 1 relative).
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a Pesa level. The MX bus levels are 0 relative. **"Pesa level"**. This is the level that will be used for the Pesa bus. The Pesa bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to a Pesa output. Outputs on the MX bus are 0 relative. **"MX output length"**. This is the output block size. **"Pesa output offset"**. This number will be added to the MX output number and this will be used as the Pesa output. This can be a positive or negative number. Outputs on the Pesa bus are 1 relative. **"Pesa input offset"**. This number will be added to the MX input number and this will be used as the Pesa input. This can be a positive or negative number. Pesa inputs are 1 relative.

PESA_LEVEL_COUNT:@<Number of levels in Pesa >

- FUNCTIONALITY - The Pesa may not accept takes with more or less levels than what they are configured for. This value is how many levels the Pesa is configured for. If this is set to 0, then the code will calculate the number of levels on power up, based on the first status response received.
- PARAMETERS - There is only one parameter. The number of levels configured in the Pesa.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

LEITCH-BUS (LEITCH Router)

The following section is for use with the Leitch router. SM stands for serial master. It is used in Master mode where the SC4 controls the Leitch routers serially. **NOTE:** Up to 6 LEITCH Terminal Serial Master routers can be defined. Simply duplicate the LEITCHTSM_BUS_DATA_START section 6 times. **NOTE:** Leitch is 1 relative, USI is 0 relative, input/output offsets will be an odd number. For levels use 0 relative for both (Leitch may be labeled 1 relative) **NOTE:** Leitch serial: common RS232,9600,8,1,N,N (note that baud rate and RS232/RS433 is switch configurable on Leitch)

[LEITCHTSM_BUS_DATA_START]
ENABLED: @TRUE or FALSE

- FUNCTIONALITY - This enables or disables the Pesa serial master bus operation.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the operation. "**FALSE**" disables the operation.

SERIAL_PORT: @<port (1 thru 6)>, <baud rate>, <bit length>,<stop bits>, <parity>, <flowcontrol>

- FUNCTIONALITY - The LEITCH bus runs on a serial port. This specifies which serial port it will run on. The serial port operating parameters are specified here. Note that the normal setting for SIERRA bus is 9600, 8 data, 1 stop, Odd parity.
- PARAMETERS - There are six parameters. **"Port"**. This is one of the six ports on the back of the MXLator. Valid numbers are 1 thru 6. **"Baud rate"**. This is the baud rate of the port. Valid baud rates are; 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200. **"Bit length"**. This can be 7 or 8. **"Stop bits"**. This can be 1 or 2. **"Parity"**. This can be O for odd, E for even, N for none. **"Flowcontrol"**. This is N for none or H for rtscts hardware handshake.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.

- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

MX_LEITCH_MAPPING: @<MX level>,<Leitch level>,<MX output start>,<MX output length>,<Leitch output offset>,<Leitch input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the Leitch will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Note that the Leitch is 1 relative and the MX is 0 relative for inputs and outputs. The levels should be mapped as 0 relative for both, (the physical Leitch router may be labeled with levels that are 1 relative).
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a Leitch level. The MX bus levels are 0 relative. **"Leitch level"**. This is the level that will be used for the Leitch bus. The Leitch bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to a HCP output. Outputs on the MX bus are 0 relative. **"MX output length"**. This is the output block size. **"Leitch output offset"**. This number will be added to the MX output number and this will be used as the Leitch output. This can be a positive or negative number. Outputs on the Leitch bus are 1 relative. **"Leitch input offset"**. This number will be added to the MX input number and this will be used as the Leitch input. This can be a positive or negative number. Leitch inputs are 1 relative.

HCP-BUS (UT100 Sandar Router)

The following section is used for Utah 100 Sandar Routers. M stands for master. It is used in Master mode where the SC4 controls the Sandar routers. **NOTE**: Up to 6 HCPM bus control routers can be defined. Simply duplicate the HCPM_BUS_DATA_START section 6 times the following fields are REQUIRED the format for this section is a FIELD LABEL followed by FIELD DATA the FIELD LABEL must have a ':' as the last character the FIELD DATA must start with a '@' char and be no longer than 20 characters. **NOTE:** Default for the ACK_WARN_TIME and ACK_CLOSE_TIME lines is 65ms. If nothing is put in these lines, they will default to 65. This number needs to be above 200ms for correct functionality.

[HCPM_BUS_DATA_START]

ENABLED: @TRUE or FALSE

- FUNCTIONALITY - This enables or disables the Pesa serial master bus operation.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the operation. "**FALSE**" disables the operation.

IP_ONE: @<IP for primary controller>

- FUNCTIONALITY - This is set to the IP address of the primary card.
- PARAMETERS – There is only one parameter. "**IP address**".

IP_TWO: @<IP for secondary controller, not required if it does not exist>

- FUNCTIONALITY – This is set to the IP address of the secondary, backup card, if there is one.
- PARAMETERS - There is only one parameter. "**IP address**".

IP_PORT: @<port number (normal is 7000)>

- FUNCTIONALITY - This is to set the ip port number for the UT100 device.
- PARAMETERS – There is only one parameter. "**7000**".

ACK_WARN_TIME: @<number of mils before ack warning>

- FUNCTIONALITY – This is to set the length of time before the acknowledge is sent.
- PARAMETERS – There is only one parameter. "**Length of milliseconds**". Minimum number 250.

ACK_CLOSE_TIME: @<number of mils before ack waiting closes connection>

- FUNCTIONALITY – This is to set the length of time in milliseconds after the acknowledge is sent before the connection closes.
- PARAMETERS – There is only one parameter. "**Length of milliseconds**". Minimum number 250.

MX_HCPM_MAPPING: @<MX level>,<HCP level>,<MX output start>,<MX output length>,<HCPM output offset>,<HCPM input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the HCP will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Note that the HCP is 1 relative and the MX is 0 relative for inputs and outputs. The levels should be mapped as 0 relative for both, (the physical HC router may be labeled with levels that are 1 relative).
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a HCP level. The MX bus levels are 0 relative. **"HCP level"**. This is the level that will be used for the HCP bus. The HCP bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to a HCP output. Outputs on the MX bus are 0 relative. **"MX output length"**. This is the output block size. **"HCP output offset"**. This number will be added to the MX output number and this will be used as the HCP output. This can be a positive or negative number. Outputs on the HCP bus are 1 relative. **"HCP input offset"**. This number will be added to the MX input number and this will be used as the HCP input. This can be a positive or negative number. HCP inputs are 1 relative.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

HCP-BUS (UT100 Sandar Routers)

The following section is used for Utah 100 Sandar Routers. S stands for slave. It is used in Slave mode where the SC4 is controlled by the Sandar routers. **NOTE:** Up to 6 hcps bus control routers can be defined. Simply duplicate the HCPS_BUS_DATA_START section 6 times the following fields are REQUIRED the format for this section is a FIELD LABEL followed by FIELD DATA the FIELD LABEL must have a ':' as the last character the FIELD

DATA must start with a '@' char and be no longer than 20 characters. **NOTE:** Default for the ACK_WARN_TIME and ACK_CLOSE_TIME lines is 65ms. If nothing is put in these lines, they will default to 65. This number needs to be above 200ms for correct functionality. **NOTE:** To use default of 1000, do not include NON_CONFIRM_TIME line.

[HCPS_BUS_DATA_START]

ENABLED: @TRUE or FALSE

- FUNCTIONALITY - This enables or disables the Pesa serial master bus operation.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the operation. "**FALSE**" disables the operation.

DESCRIPTION: @<Description of this device used over HCP>

- FUNCTIONALITY - This is to describe the way this device will function.
- PARAMETERS - There is only one parameter. "Type the description."

IP_PORT: @<port number (normal is 7000)>

- FUNCTIONALITY - This is to set the ip port number for the UT100 device.
- PARAMETERS – There is only one parameter. "**7000**".

ACK_WARN_TIME: @<number of mils before ack warning>

- FUNCTIONALITY – This is to set the length of time before the acknowledge is sent.
- PARAMETERS – There is only one parameter. "**Length of milliseconds**". Minimum number 250.

ACK_CLOSE_TIME: @<number of mils before ack waiting closes connection>

- FUNCTIONALITY – This is to set the length of time in milliseconds after the acknowledge is sent before the connection closes.
- PARAMETERS – There is only one parameter. "**Length of milliseconds**". Minimum number 250.

MX_HCPS_MAPPING: @<MX level>,<HCP level>,<MX output start>,<MX output length>,<HCPM output offset>,<HCPM input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the HCP will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Note that the HCP is 1 relative and the MX is 0 relative for inputs and outputs. The levels should be mapped as 0 relative for both, (the physical HC router may be labeled with levels that are 1 relative).
- PARAMETERS - There are six parameters. **"MX level"**. This level on the MX bus will be mapped by the MXLator to a HCP level. The MX bus levels are 0 relative. **"HCP level"**. This is the level that will be used for the HCP bus. The HCP bus levels are 0 relative. **"MX output start"**. This is the first output on the MX bus that will map to a HCP output. Outputs on the MX bus are 0 relative. **"MX output length"**. This is the output block size. **"HCP output offset"**. This number will be added to the MX output number and this will be used as the HCP output. This can be a positive or negative number. Outputs on the HCP bus are 1 relative. **"HCP input offset"**. This number will be added to the MX input number and this will be used as the HCP input. This can be a positive or negative number. HCP inputs are 1 relative.

NON_CONFIRM_TIME: @<Time to wait before non-confirm send, in mil seconds>

- FUNCTIONALITY -
- PARAMETERS -

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it's set to RCP then the RCP-3 section will also need to be configured.
- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

LAWO-BUS (LAWO Router)

The following section is used for Lawo routers. NM stands for Network Master. It is used in Master mode where the SC4 controls the Lawo routers. **NOTE:** Up to 6 Lawo network master bus control routers can be defined. Simply duplicate the LWONM_BUS_DATA_START section 6 times the following fields are REQUIRED the format for this section is a FIELD LABEL followed by FIELD DATA the FIELD LABEL must have a ':' as the last character the FIELD DATA must start with a '@' char and be no longer than 20 characters.

[LAWONM_BUS_DATA_START]

ENABLED: @TRUE or FALSE

- FUNCTIONALITY - This enables or disables the Pesa serial master bus operation.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the operation. "**FALSE**" disables the operation.

IP_ONE: @<IP for primary controller>

- FUNCTIONALITY - This is set to the IP address of the primary card.
- PARAMETERS – There is only one parameter. "**IP address**".

IP_TWO: @<IP for secondary controller, not required if it does not exist>

- FUNCTIONALITY – This is set to the IP address of the secondary, backup card, if there is one.
- PARAMETERS - There is only one parameter. "**IP address**".

IP_PORT: @<port number>

- FUNCTIONALITY – This is the IP port for each of the Lawo routers.
- PARAMETERS – There is only one parameter. "**IP Port**".

MAPPING_TABLE: @<mapping table index 0 to 15>

- FUNCTIONALITY - This is to set the mapping index number for each of the Lawo routers used.

- PARAMETERS – There is only one parameter. “**Lawo index router number**”.

DATA_REFRESH_TIME: @<number of mils between refresh requests>

- FUNCTIONALITY – This is the amount of time in milliseconds between each refresh request.
- PARAMETERS – There is only one parameter. “**Time in milliseconds**”.

MX_LWONM_MAPPING: @<MX level>,<MX output start>,<MX output length>,<LWONM output offset>,<LWONM input offset>

- FUNCTIONALITY - This specifies the blocks of outputs that the LAWO will map to, from the MX bus. This block of outputs must be included in the [MX_BUS_DATA_START] LEVEL_MAPPING configuration block of outputs. Note that the LAWO is 1 relative and the MX is 0 relative for inputs and outputs. The levels should be mapped as 0 relative for both, (the physical LAWO router may be labeled with levels that are 1 relative).
- PARAMETERS - There are six parameters. **“MX level”**. This level on the MX bus will be mapped by the MXLator to a LAWO level. The MX bus levels are 0 relative. **“LAWO level”**. This is the level that will be used for the LAWO bus. The LAWO bus levels are 0 relative. **“MX output start”**. This is the first output on the MX bus that will map to a LAWO output. Outputs on the MX bus are 0 relative. **“MX output length”**. This is the output block size. **“LAWO output offset”**. This number will be added to the MX output number and this will be used as the LAWO output. This can be a positive or negative number. Outputs on the LAWO bus are 0 relative. **“LAWO input offset”**. This number will be added to the MX input number and this will be used as the LAWO input. This can be a positive or negative number. LAWO inputs are 0 relative.

USI_SOURCE:@< MX | RCP >

- FUNCTIONALITY - The source of takes from the SC4 can be defined. Two options exist, MX bus or RCP3/4. If it’s set to RCP then the RCP-3 section will also need to be configured.

- PARAMETERS - There is only one parameter. The source of the SC4 takes. **MX** or **RCP**.

MXLator-LOGGING

The following section is for use when additional logging information is required. There is a debug logging section of the MXlator that can be used to isolate switches and capture event logs.

[LOGGING_DATA_START]

LOGGING_SNIFFER_ENABLED:@< TRUE | FALSE >

- FUNCTIONALITY - This enables or disables the sniffer mode. None of the other functions are executed in sniffer mode. The only other configuration that is used is the MX router. If the MX router is enabled, the configured section will be simulated on the MX bus.
- PARAMETERS - There is only one parameter. "**TRUE**" enables the Sniffer. "**FALSE**" disables the Sniffer.

SERVER_NAME:@<logging server name>

- FUNCTIONALITY - This is the logging server name that will be used. If this is not specified, "USI_SYSLOG01" will be used.
- PARAMETERS - There is only one parameter. The name of the logging server.

LOG_COMMAND:@<debug command>

- FUNCTIONALITY - This allows debug commands at power up to be saved in the config.
- PARAMETERS - There is only one parameter. The debug command line, same as you would type it.

Burning the MXLator Configuration File

Perform the following steps to program the configuration file into the MXLator.

1. Locate and launch the NFS program. A shortcut may be found on the desktop in a folder called 'Utah Shortcuts'. If not then it will for sure be located in Program Files in the start menu and is called ProNFS.
2. Locate and launch the Utah Install Utility which is generally found in the C:\usi directory. A shortcut may be found on the desktop in a folder called 'Utah Shortcuts' (see Figure 32).

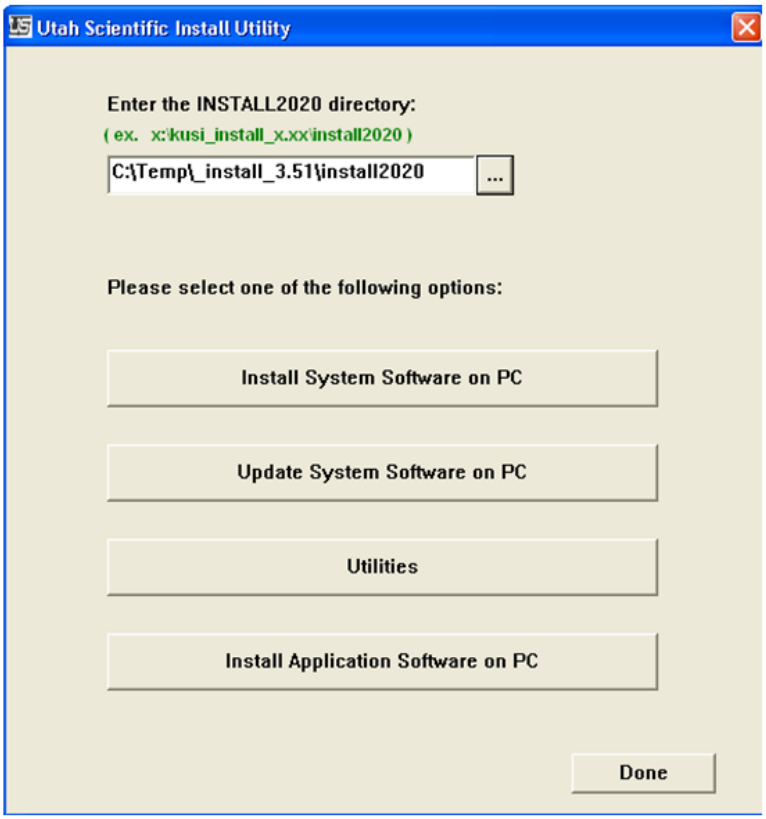


Figure 3-32.

3. Click the ‘Utilities’ button (figure 33) and then drop down the ‘Device Type Filter’ menu (figure 34) and select MXLATOR from the list.

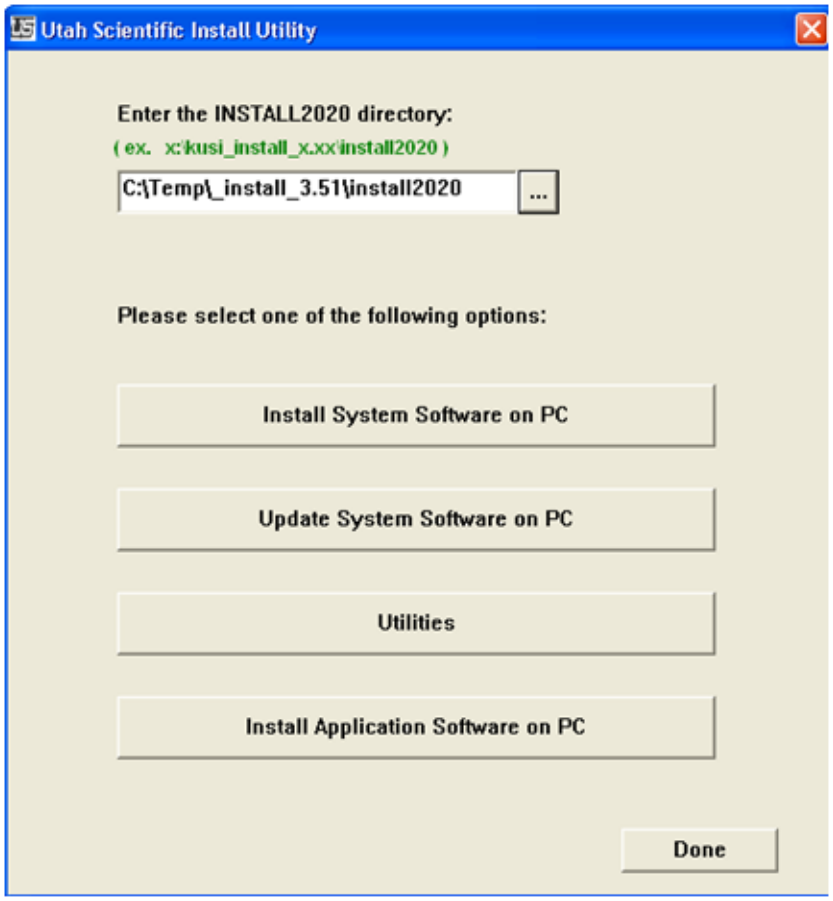


Figure 3-33.

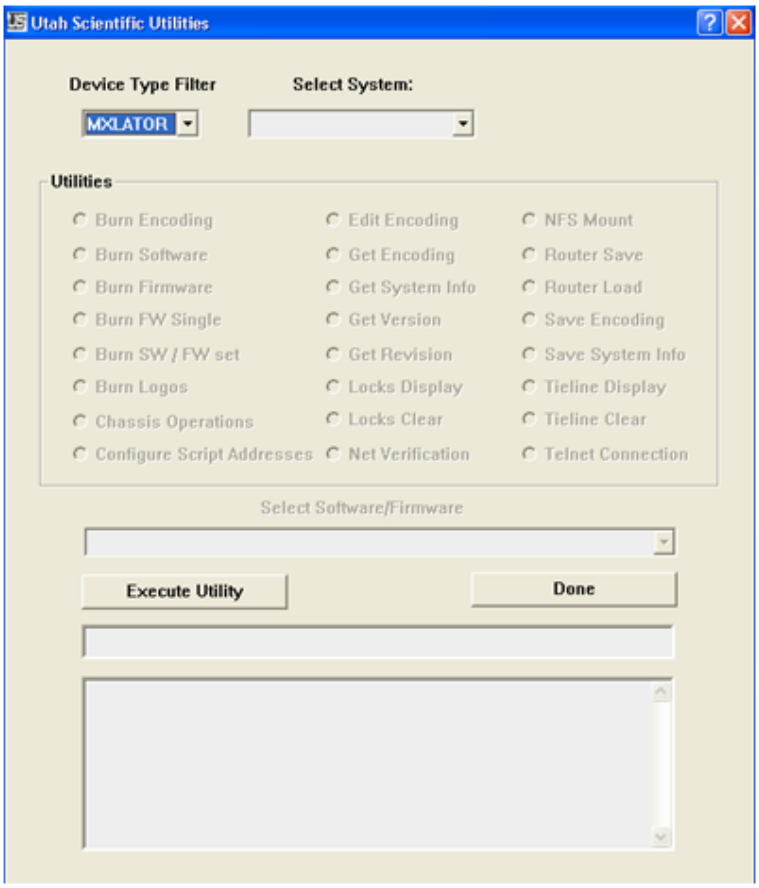

Utah Scientific Utilities
Device Type Filter
Select System:
MXLATOR
Utilities
Burn Encoding
Edit Encoding
NFS Mount
Burn Software
Get Encoding
Router Save
Burn Firmware
Get System Info
Router Load
Burn FW Single
Get Version
Save Encoding
Burn SW / FW set
Get Revision
Save System Info
Burn Logos
Locks Display
Tieline Display
Chassis Operations
Locks Clear
Tieline Clear
Configure Script Addresses
Net Verification
Telnet Connection
Select Software/Firmware
Execute Utility
Done

Figure 3-34.

4. Click on the ‘Select System’ drop down menu and then click on the file called ‘xxxx-mxlator01’ (xxxx is short for the call letters). See Figure 35 below.

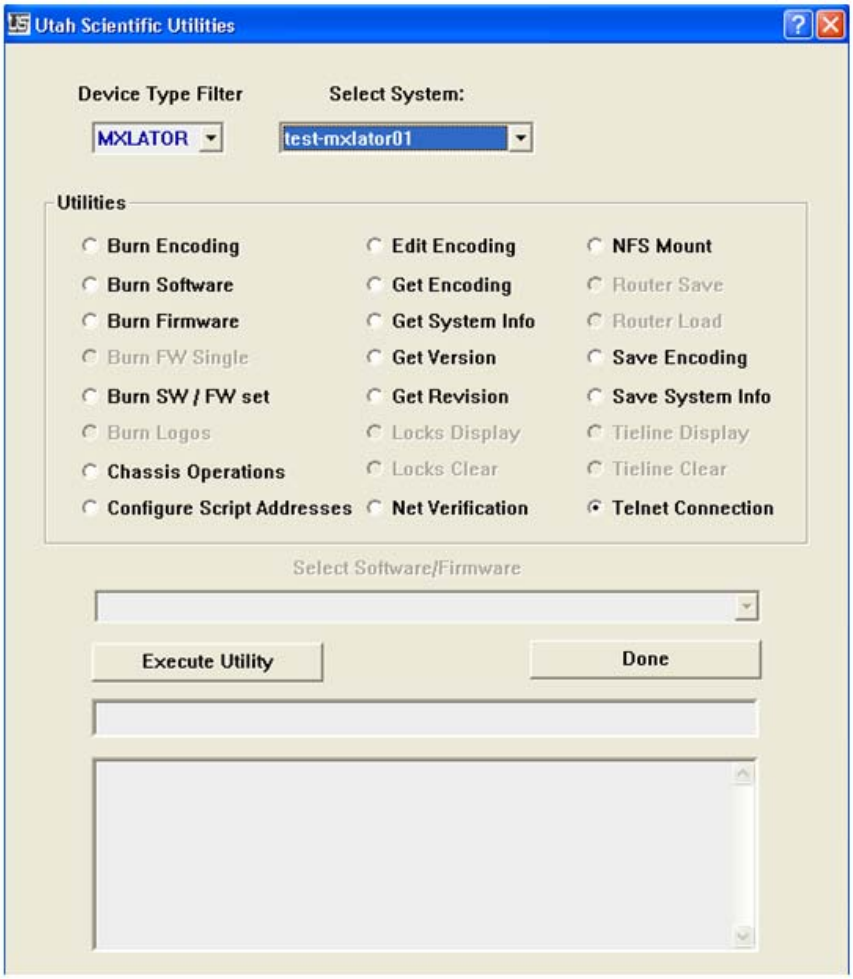


Figure 3-35.

5. Click on the 'Burn Encoding' (figure 36) item on the main screen and then click the 'Execute Utility' button (figure 37) in the lower half of the utility screen. The programming should complete with a green message in the message bar below the Execute Utility button stating 'Operation Completed Successful'.

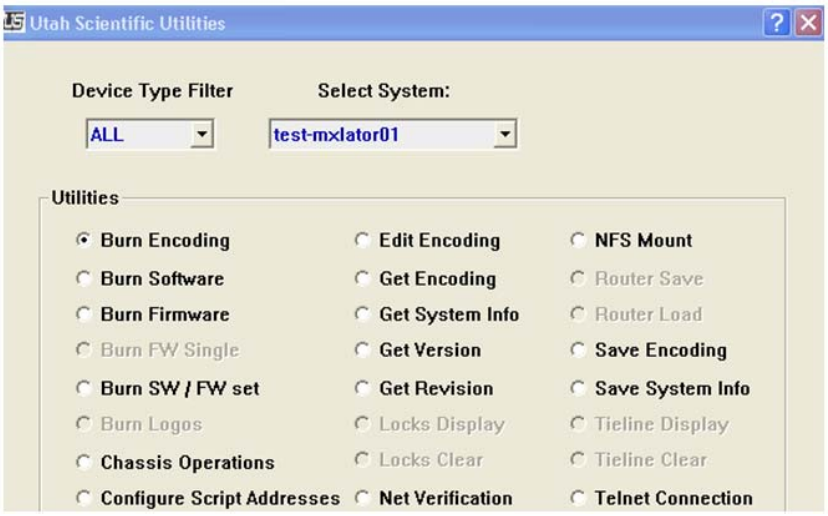


Figure 3-36.

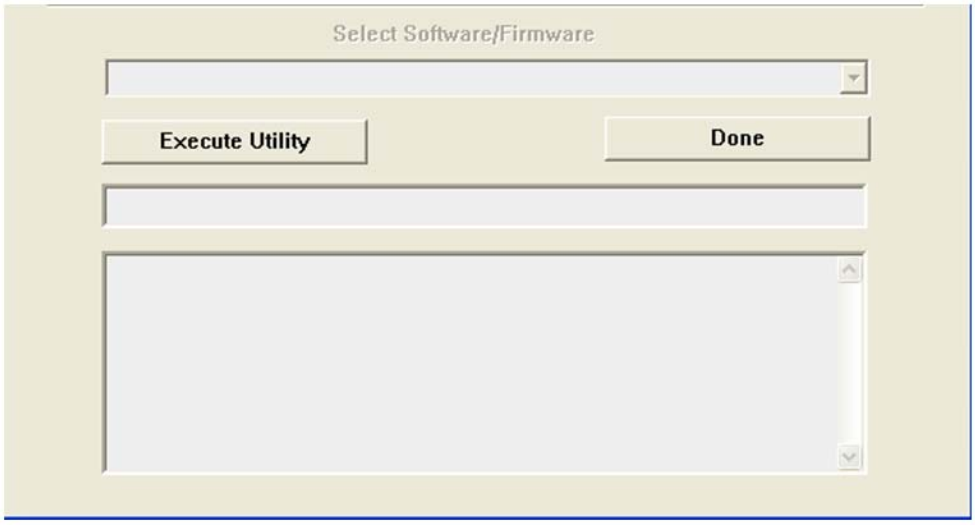


Figure 3-37.

Example Config Files (Including known issues)

GVG-BUS (SMS-7000 CONTROLLED BY UTAH)

[MX_BUS_DATA_START]

ENABLED: @TRUE

READ_BACK_STATUS: @1

READ_BACK: @0x0060

DISCONNECT_INPUT: @240

LEVEL_MAPPING: @1, 0, 128

[RCP3_BUS_DATA_START]

ENABLED: @TRUE

IP: @192.000.002.100

REFRESH_TIME: @200

REFRESH_BLOCK: @8

[GVNMN_BUS_DATA_START]

ENABLED: @TRUE

IP_ONE: @192.0.2.2

IP_TWO: @192.0.2.3

MX_GVNMN_MAPPING: @1, 0, 00,128, 0, 0

USI_SOURCE: @MX

PESA-BUS (PESA CONTROLLED BY UTAH)

[MX_BUS_DATA_START]

ENABLED: @TRUE

READ_BACK_STATUS: @1

READ_BACK: @0x0060

DISCONNECT_INPUT: @300

LEVEL_MAPPING: @3, 0, 288

LEVEL_MAPPING: @4, 0, 288

LEVEL_MAPPING: @5, 0, 288

LEVEL_MAPPING: @6, 0, 288

LEVEL_MAPPING: @7, 0, 288

LEVEL_MAPPING: @8, 0, 288

LEVEL_MAPPING: @9, 0, 288

LEVEL_MAPPING: @10, 0, 288

LEVEL_MAPPING: @11, 0, 288

LEVEL_MAPPING: @12, 0, 288

LEVEL_MAPPING: @13, 0, 288

LEVEL_MAPPING: @14, 0, 288

LEVEL_MAPPING: @15, 0, 288

[PESASM_BUS_DATA_START]

ENABLED: @TRUE

SERIAL_PORT: @1, 38400, 8, 2, N , H

MX_PESA_MAPPING: @3, 0 , 0, 288, 0, 0

MX_PESA_MAPPING: @4, 1 , 0, 288, 0, 0

MX_PESA_MAPPING: @5, 2 , 0, 288, 0, 0

MX_PESA_MAPPING: @6, 3 , 0, 288, 0, 0

MX_PESA_MAPPING: @7, 4 , 0, 288, 0, 0

MX_PESA_MAPPING: @8, 5 , 0, 288, 0, 0

MX_PESA_MAPPING: @9, 9 , 0, 288, 0, 0

MX_PESA_MAPPING: @10, 10 , 0, 288, 0, 0

MX_PESA_MAPPING: @11, 11 , 0, 288, 0, 0

MX_PESA_MAPPING: @12, 12 , 0, 288, 0, 0

MX_PESA_MAPPING: @13, 13 , 0, 288, 0, 0

MX_PESA_MAPPING: @14, 14 , 0, 288, 0, 0

MX_PESA_MAPPING: @15, 15 , 0, 288, 0, 0

USI_SOURCE: @MX

PESA_LEVEL_COUNT: @0

PROBEL (PROBEL BUS CONTROLLING UTAH)

[RCP3_BUS_DATA_START]

ENABLED: @TRUE

IP: @192.168.005.230

REFRESH_TIME: @150

REFRESH_BLOCK: @3

[PROBELNS_BUS_DATA_START]

ENABLED: @TRUE

IP_PORT_PROTOCOL: @2000

IP_PORT_ERRORS: @2002

DEFAULT_SOURCE: @700

MX_PROBEL_MAPPING: @1, 0, 0, 300, 0, 0

PROBEL ROUTER (PROBEL BUS CONTROLLED BY UTAH)

[MX_BUS_DATA_START]

ENABLED: @TRUE

READ_BACK_STATUS: @1

READ_BACK: @0x0060

DISCONNECT_INPUT: @129

LEVEL_MAPPING: @1, 1, 128

LEVEL_MAPPING: @3, 1, 64

[PROBELSM_BUS_DATA_START]

ENABLED: @TRUE

SERIAL_PORT: @5, 38400, 8, 1, E , N

MX_PROBEL_MAPPING: @1, 0, 128, 0, 0

USI_SOURCE: @MX

AVS ROUTER (DATA BUS CONTROLLED BY UTAH)

[MX_BUS_DATA_START]

ENABLED: @TRUE

READ_BACK_STATUS: @1

READ_BACK: @0x0060

DISCONNECT_INPUT: @30

LEVEL_MAPPING: @0, 0, 120

LEVEL_MAPPING: @1, 0, 100

LEVEL_MAPPING: @2, 0, 50

[DB_BUS_DATA_START]

ENABLED: @TRUE

320_OUTPUTS: @FALSE

25_DATA_BITS: @FALSE

MX_DB_MAPPING: @0, 0, 0, 120, 0, 0

MX_DB_MAPPING: @1, 1, 0, 100, 0, 0

MX_DB_MAPPING: @2, 2, 0, 50, 0, 0

REFRESH_LEVEL: @0

REFRESH_LEVEL: @1

REFRESH_LEVEL: @2

REFRESH_DECADE: @0

REFRESH_DECADE: @1

REFRESH_DECADE: @2

REFRESH_DECADE: @3

REFRESH_DECADE: @4

USI_SOURCE: @MX

VENUS ROUTER

(Venus ES Bus controlled by Utah but still able to use the Venus Panels to switch the same outputs on the Venus routers only and not switching the Utah routers)

The following sections came from a working config file where the Venus router outputs were switched by both the Utah control panels and the GV (Phillips) control panels. The Utah router was only switched by its own panels and not the GV panels. The other requirement was that refresh was turned off so that all outputs could be in the router tables and not refresh the Venus outputs. This protocol section asks for updates from the VM3000 and replies to the SC4 over the RCP3 ethernet.

For continuous refresh to be on, it would require the section titled ESBCM_BUS_DATA_START to be used instead of this section. If the Utah router also needs to be switched by the GV panels then this section would also require the MX_ESBC_LCL_MAPPING to be filled in as well as the SERIAL_PORT_TR.

The following items are required in order to establish a connection and switching of the Venus routers from the MXLator.

1. In Jupiter Control System create new line as a serial device in the MPK Devices table.
2. Create a new serial input set.
3. Create a new serial output set.
4. Choose or create a level set. This will show names of routers next to row numbers. Use the row number, minus 1, as the level number in the config file MX_ESBC_MAPPING line found in the protocol section in the MXLator config file as seen below.

5. Choose a port on the VM3000 controller in the serial protocol settings. If all the ports are used up then there should be a serial expansion unit called an SI3000 to use. Note: For older systems the ports in this table will be labeled in pairs such as 1/2, 3/4, 5/6, 7/8 and so forth. These are physical ports on the rear of the VM or SI 3000 unit and each pair are shared on the same IC on the internal circuit board. For newer systems the port is labeled a single port such as (COM 1, COM 2, etc).
6. Choose protocol called ES-Switch in serial protocol settings.
7. Create a serial cable with the following pin configuration.

[MX_BUS_DATA_START]

ENABLED: @TRUE

READ_BACK_STATUS: @1

READ_BACK: @0x0060

DISCONNECT_INPUT: @4095

LEVEL_MAPPING: @1, 0, 64

LEVEL_MAPPING: @2, 0, 64

LEVEL_MAPPING: @3, 0, 64

[ESBC_BUS_DATA_START]

ENABLED: @TRUE

SERIAL_PORT: @1, 38400, 8, 1, O, N (8,1,O,N is always the case – baud can vary)

MX_ESBC_MAPPING:@1, 0, 0, 64, 0, 0 (2nd column is Venus level set row number minus 1)

MX_ESBC_MAPPING:@2, 1, 0, 64, 0, 0 (2nd column is Venus level set row number minus 1)

MX_ESBC_MAPPING:@3, 2, 0, 64, 0, 0 (2nd column is Venus level set row number minus 1)

USI_SOURCE: @MX (always MX)

[RCP3_BUS_DATA_START]

ENABLED: @TRUE

IP: @10.13.96.162 (this is the ip add of the SC4)

REFRESH_TIME: @150 (default – leave at 150)

REFRESH_BLOCK: @3 (default – leave at 3)

VENUS ROUTER (VENUS ES BUS CONTROLLED BY UTAH ONLY)

The following sections came from a working config file where the Venus router outputs were switched by the Utah control panels only. This protocol doesn't allow the Venus panels to switch the same outputs that are set up in the router output tables. The refresh is continuously on so all outputs that are in the output range and encoded in the SC4 will refresh. Any outputs that are not wanted to be switched by Utah need to be removed from the router output table for those levels.

The following items are required in order to establish a connection and switching of the Venus routers from the MXLator.

1. In Jupiter Control System create new line as a serial device in the MPK Devices table.
2. Create a new serial input set.
3. Create a new serial output set.
4. Choose or create a level set. This will show names of routers next to row numbers. Use the row number, minus 1, as the level number in the config file MX_ESBC_MAPPING line found in the protocol section in the MXLator config file as seen below.
5. Choose a port on the VM3000 controller in the serial protocol settings. If all the ports are used up then there should be a serial expansion unit called an SI3000 to use. Note: For older systems the ports in this table will be labeled in pairs such as 1/2, 3/4, 5/6, 7/8 and so forth. These are physical ports on the rear of the VM or SI 3000 unit and each pair are shared on the same IC on the internal circuit board. For newer systems the port is labeled a single port such as (COM 1, COM 2, etc).
6. Choose protocol called ES-Switch in serial protocol settings.
7. Create a serial cable with the following pin configuration.

[MX_BUS_DATA_START]

ENABLED: @TRUE

READ_BACK_STATUS: @1

READ_BACK: @0x0060

DISCONNECT_INPUT: @4095

LEVEL_MAPPING: @2, 0, 64

LEVEL_MAPPING: @3, 0, 64

LEVEL_MAPPING: @4, 0, 64

ESBCM_BUS_DATA_START]

ENABLED: @TRUE

SERIAL_PORT: @3, 38400, 8, 1, O, N (8,1,O,N is always the case – baud can vary)

MX_ESBCM_MAPPING:@2, 0, 0, 64, 0, 0 (2^{nd} column is Venus level set row number minus 1)

MX_ESBCM_MAPPING:@3, 1, 0, 64, 0, 0 (2^{nd} column is Venus level set row number minus 1)

MX_ESBCM_MAPPING:@4, 2, 0, 64, 0, 0 (2^{nd} column is Venus level set row number minus 1)

USI_SOURCE: @MX

[RCP3_BUS_DATA_START]

ENABLED: @FALSE

IP: @192.168.038.021

REFRESH_TIME: @150

REFRESH_BLOCK: @3

VENUS (ES BUS CONTROLLING UTAH)

[MX_BUS_DATA_START]

ENABLED: @TRUE

READ_BACK_STATUS: @1

READ_BACK: @0x0060

DISCONNECT_INPUT: @30

LEVEL_MAPPING: @1, 0, 272

[ESTR_BUS_DATA_START]

ENABLED: @TRUE

SERIAL_PORT: @1, 38400, 8, 1, O, N

MX_ESTR_MAPPING:@1, 1, 0, 272, 0, 0

USI_SOURCE: @MX

[RCP3_BUS_DATA_START]

ENABLED: @TRUE

IP: @192.168.005.230

REFRESH_TIME: @150

REFRESH_BLOCK: @3

UTAH 100 ROUTER (HCPM BUS CONTROLLED BY UTAH)

[MX_BUS_DATA_START]

ENABLED: @TRUE

READ_BACK_STATUS: @1

READ_BACK: @0x0060

DISCONNECT_INPUT: @100

LEVEL_MAPPING: @3, 0, 32

LEVEL_MAPPING: @4, 0, 32

[HCPM_BUS_DATA_START]

ENABLED: @TRUE

IP_ONE: @010.009.038.004

IP_PORT: @7000

ACK_CLOSE_TIME:@350

ACK_WARN_TIME:@350

MX_HCPM_MAPPING:@4, 1, 0, 32, 0, 0

USI_SOURCE: @MX

[HCPM_BUS_DATA_START]

ENABLED: @TRUE

IP_ONE: @010.009.038.005

IP_PORT: @7000

ACK_CLOSE_TIME:@350

ACK_WARN_TIME:@350

MX_HCPM_MAPPING:@3, 1, 0, 32, 0, 0

USI_SOURCE: @MX

AVS-2 ROUTER (SC BUS CONTROLLED BY UTAH)

[MX_BUS_DATA_START]

ENABLED: @TRUE

READ_BACK_STATUS: @1

READ_BACK: @0x60

```
DISCONNECT_INPUT:    @4095
LEVEL_MAPPING:       @4, 0, 160
LEVEL_MAPPING:       @5, 0, 160
LEVEL_MAPPING:       @6, 0, 160
LEVEL_MAPPING:       @7, 0, 160

[SC_BUS_DATA_START]
ENABLED:             @TRUE
MX_SC_MAPPING:       @4, 0, 0, 160, 0, 0
MX_SC_MAPPING:       @5, 1, 0, 160, 0, 0
SCDR_AUTO_DISCONNECT:@TRUE
SCDR_EXCLUSIVE_LOCK:@FALSE
REFRESH_LEVEL:@0
REFRESH_LEVEL:@1
REFRESH_DECADE:@0
REFRESH_DECADE:@1
REFRESH_DECADE:@2
REFRESH_DECADE:@3
REFRESH_DECADE:@4
```

ROSS PRODUCTION SWITCHER (CONTROLLED BY ROSS)

(Used at DMA)

The following sections came from a working config file where the Ross Production Switcher reads back the entire list of sources from the SC4/SC400 controller as well as switching those sources to designated outputs.
The following items are required in order to establish a connection and correct switching and status of the Ross Production Switcher, which uses the GV protocol.

1. The Ross panel and frame must be on the same subnet as the MXLator. LAN 2 on the Ross frame is only available on version 12 or higher, otherwise there is only use of LAN 1.
2. A 'Router' must be enabled on the Ross. If not, then a phone call to Ross from the customer is required to get a code to enable it.
3. To set up communications on the Ross:
   a. go to 'Home Screen – More – Setup – Installation – Comm Setup'
   b. Set the Comm port to 'Remote 1' or first available if Remote 1 is in use.
   c. Set 'Type' to 'Router'.
   d. Go to the bottom and push 'Select Device'. Choose 'GVGNP' from right drop down window.
   e. Go to the bottom and push 'Comm Type'. Set to 'Network TCP'.
   f. Go to the bottom and push 'Comm Settings'. In the 'Client Server' window select 'Client'.
   g. Enter the remote IP address of the MXLator and the remote port from the MXLator config file which should be '12345'. Leave the local IP and local port blank with 0's.
   h. Go to 'Home' and confirm changes.
4. Go to the 'BNC Table' at the bottom of the screen. 'Home – More – Setup – Installation – BNC – BNC Type.
   a. Assign each BNC at the right of the screen as 'Router' in the 'Type' drop down list.

b. Push ‘More – Router Setup’. Assign BNC outputs at the right, to the BNC output it is assigned to on the Utah Scientific router. This is the physical BNC on the router.
c. Go to ‘Home’ and confirm changes.

[RCP3_BUS_DATA_START]

ENABLED: @TRUE

IP: @192.168.005.230

REFRESH_TIME: @150

REFRESH_BLOCK: @3

[GVNSN_BUS_DATA_START]

ENABLED: @TRUE

PORT: @12345

LEVEL_4_ACK: @TRUE

TIME_OUT: @0

MX_GVNSN_MAPPING: @0,0,0,144,0,0

USI_SOURCE: @RCP

MXLATOR Console for Serial Port Status

The console mode can be enabled from the diagnostics port on the active MXLator card or using a Telnet connection to it. The preferred method is the serial connection to the diagnostics port as this will also read out serial traffic with no intervention.

1. Using a terminal program such as TeraTerm, connect to the MXLator. For a serial connection, connect to the front port using the RJ45 to 9 pin adapter that was provided with the system labeled 'SC4/2020'. The baud rate is 19200 and the parity is 8N1.
2. At the prompt, type ps. This will display the services running on the MXLator.
3. Find the service in the list called mxlator and note the number for that item.
4. At the prompt, type kill(space)(number from step 3 above).
5. Type ps again to note that the mxlator item is no longer in the list.
6. At the prompt, type the word mxlator. This will put you into the console mode and you will see a blank line each time the return key is pressed.
7. At the blank line, type a ? followed by the return key to see all the help items that can be used in the console.
8. At the blank prompt, type mapping and then the return key to display all the mapping tables currently in the MXLator.
9. At the blank line, type console on. If the serial connection is connected correctly there should be information on the screen depending on the type of protocol that is being used. With any of the protocols, if a take is made to switch any routers, there should be confirmed takes with legible source and destination information that are reported on the screen.
10. To turn off the console type console off.
11. To exit the console, hold the control key down and press the letter C.

12. When finished using the console, reset the MXLator card by pressing the reset button on the active MXLator card, noted by the Active LED on the card.

4

Chapter 4

Operations

This Chapter Contains the Following:

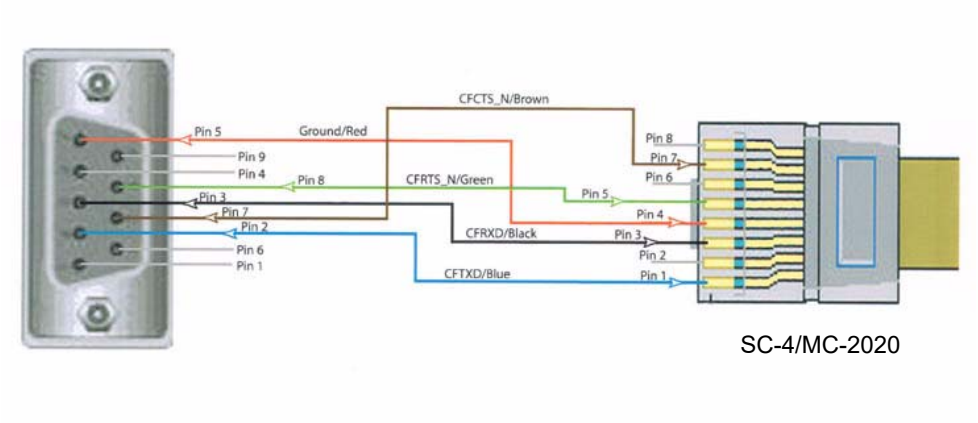
The Debug Port

Introduction

The debug port of the SC-400 is the major source of operational information and control. While configuration info is downloaded to the SC-400 via UCON, some operational information is still best determined by a direct connection to the debug port.

Physical Connection

Supplied with your SC-400 is an adapter intended to plug on to the serial port of any standard PC. This adapter is USI PN 140100-2. It has a male 9 pin D connector on one end and a female RJ-45 connector on the other. The user supplies a straight thru CAT 5 cable between the adapter and the debug port on the SC-400 (see figure 3-1 in this manual for location).



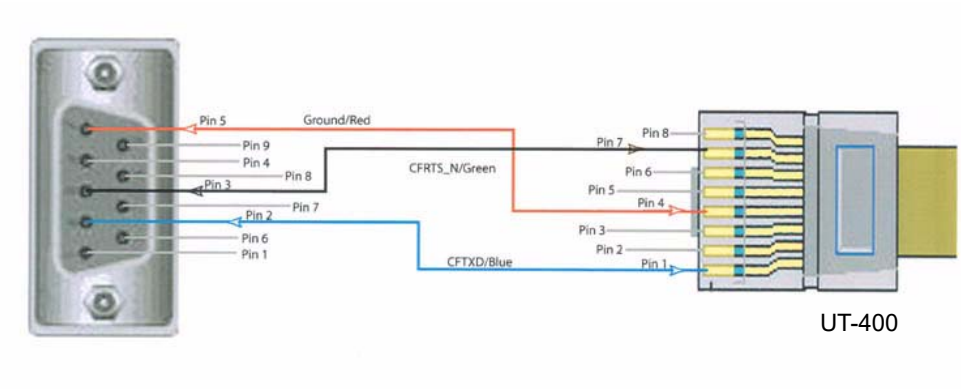


Figure 4-1. RJ-45 Pinout detail

Terminal Emulation software setup

Any terminal emulation program such as windows Hyper terminal is useful for debug port operation. Utah Scientific recommends TeraTerm, a freeware terminal emulation software package available on the internet. Serial Port parameters necessary to communicate to the SC-400 are as follows:

TABLE 4-1.

Baud Rate	19.2K
Data Bits	8
Stop Bits	1
Parity	NONE
Hardware Handshaking	NONE

General Status

Once you are connected to the SC-400, pressing the return key should return a prompt of "/>". From this prompt various operations can be performed.

The most valuable data is gathered after a system reset. This is the point where the SC-400 'finds' all connected routers and control panels. The best method for determining what the SC-400 discovers is to connect the debug port to the upper SC-400 and then press and release the reset buttons (See figure 3-1) simultaneously. After several screens of data that deal with the bootup process, you will see messages such as the following:

MX H/W found: input 0, output 0, level 0, id 0x40

configServer: register PL panel 3, name CSP-500, model CSP-16160

MX H/W messages indicate that the SC-400 has found and MX-Bus based router at the specified address. The ID number varies with different address types.

configServer messages indicate panel types and address found.

Specific Operations

Reading system setup information

The chassis command is used to read and/or set the chassis IP address(s), system name and other parameters. It's usage is detailed below.

chassis

Synopsis: chassis [OPTIONS] [FILE]

This program allows the user to interact with the chassis identification ROM located in the local chassis. Not all new USI products support this feature. If you aren't sure or are getting unexpected results, please check the appendix and verify the operation is supported on the platform you are working with.

Options:

- -h – displays the help information
- -d – activates debug mode which displays low level interactions with the device hardware.
- -r – read and display data contained in the local chassis config ROM.
- -init – wirte a set of default values into the chassis config ROM
- -ip1 <xxx.xxx.xxx.xxx> – sets the IP address for network interface 1 to the specified value.
- -ip2 <xxx.xxx.xxx.xxx> – sets the IP address for network interface 2 to the specified value.
    - Please note not all devices support a 2nd network interface.
- -cust <customer name> - Sets the customer name to the text specified. Typically this will be the call letters of the station using the device.
- -chid <chassis ID> - This sets the chassis ID to the text specified. Typically this will be a description of the physical device location. Ex: MCP-CR01-KUSI
- -unet <xxx> - set the unet node for this device to that specified.

Shutting down the SC-400 application

The SC-400down command will turn off the SC-400 application. This command might be used prior to reprogramming the SC-400's main image.

SC-400down

Synopsis: SC-400down

This command sends a signal to all active SC-400 processes and tells them to shutdown. This replaces the need to perform a <ps> command followed by a kill –INT <pid> command which is the method for stopping the current SC-400 software from running. A message will tell the user whether the shutdown has been successful or not.

Resetting the SC-400

SC-400reset

Synopsis: SC-400reset

This command resets the SC-400 board being accessed. When this action is performed on an active board in a redundant system, control will switch over to the other board.

Gathering data from the onboard log

logdump

Synopsis: logdump [OPTIONS]

This command displays log entries from NVRAM. If no option is specified then ALL log entries will be dumped to the screen.Optionally, a number can follow the command, which specifies that only that number of final entries will be displayed. The log can only handle a certain number of entries. Older entries will be lost as the log overflows saving only the most recent events.

Example:logdump 10 - dumps the last ten log messages to the screen

Viewing / Setting Configuration parameters

SC-400config is used to display and in some cases modify the custom configuration parameters of the SC-400. Various options are detaled below.

SC-400config

SC-400config sim – Turns on simulation for all router levels. Append a 1 to simulate only level 1

SC-400config nosim – Turns off simulation for all router levels. Append a 1 to unsimulate only level 1

SC-400config mx – Displays MX Bus configuration

SC-400config pl – Displays Party Line Configuration

SC-400config routers – Displays router configuration

SC-400config levels – Displays level mapping

SC-400config srcs – Displays source table

SC-400config dsts – Displays destination table

SC-400config serial – Displays serial port parameters

SC-400config misc – Displays miscellanous parameters

tiedump

Displays the tieline status

tiedump_clear – clears out the tieline status (crosspoints are not affected.) This will free all tie lines.

lockdump

This displays the output lock status.

lockdump_clear – removes all locks.

5

Chapter 5

Global Operations

In this Chapter:

Global Routing has been developed as a means of controlling multiple routers, their corresponding crosspoints and global tie lines. Global Routing can function within two basic operational scenarios; where a single, or potentially multiple MX-Lators are involved.

The ‘Global Client’ is a software interface that manages all SC-4 and SC-400 interaction. A system will contain a separate ‘client’ for each SC-4/SC-400. Global Clients are software packages run locally on the MX-Lator chassis.

The ‘Global Process’ manages global tie-lines and machine-to-machine controller communications. While a system can contain multiple ‘clients’, it will only contain one ‘Global Process’. The Global Process provides the user the ability to switch a router’s crosspoints from a remote location. The Global Process also gives an operator the ability to manage all global tie-lines.

Whether under one roof or across great distances grographically, Global Routing operations are expandable to as many MX-Lator chassis as needed. Multiple MX-Lator chassis are configured through the use of a local area network (LAN).

Operational configurations for Global Routing can be very unique, as some conditions may require a single *process* and two *clients*[1], while others might require multiple clients.

The following illustrations (in development) contain several basic operational scenarios.

Specific detail regarding software and hardware configuration will follow in a later section.

1. One Global Process with a minimum ot two clients.

Operational Overview

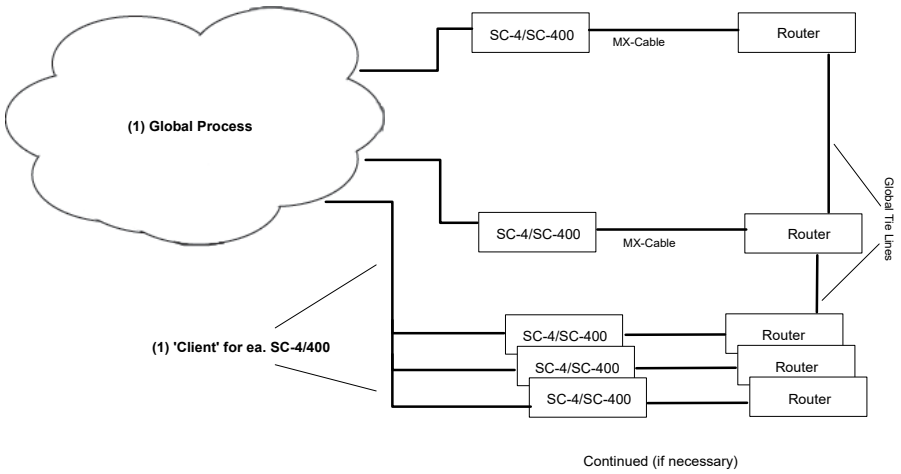


Figure 5-1. Operational Overview

The Utah-400 routers represent the hardware portion, while the SC-4 and SC-400 represent the softare side of the operation. ‘Client’ and ‘Global Process’ are also considered software representations,and can operate on a single MX-Lator chassis (in one location), or on multiple MX-Lator chassis in remote operations.

Global Processes - Two Geographic Locations

Option A

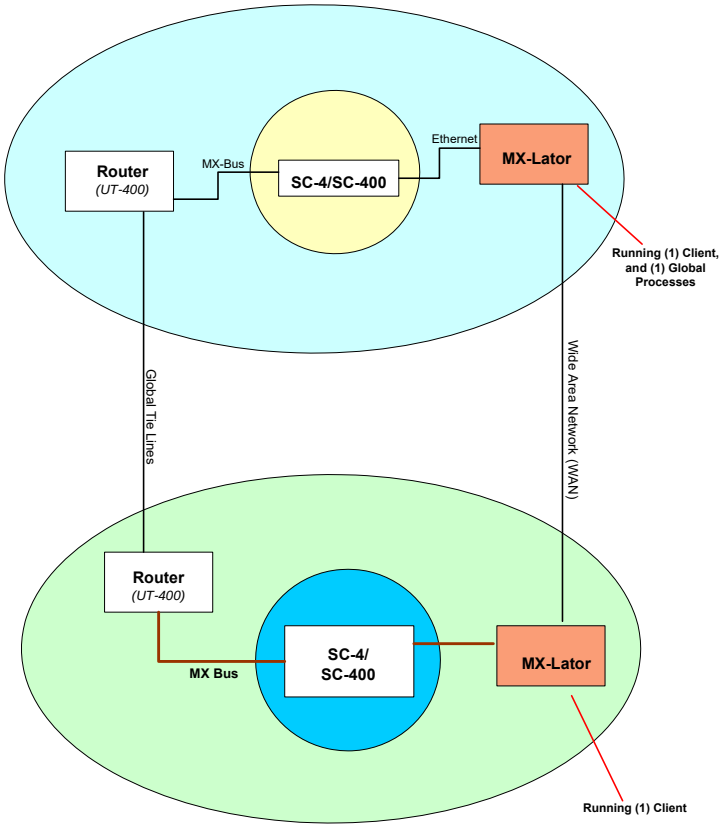


Figure 5-2. Two Locations - Option A

Option B

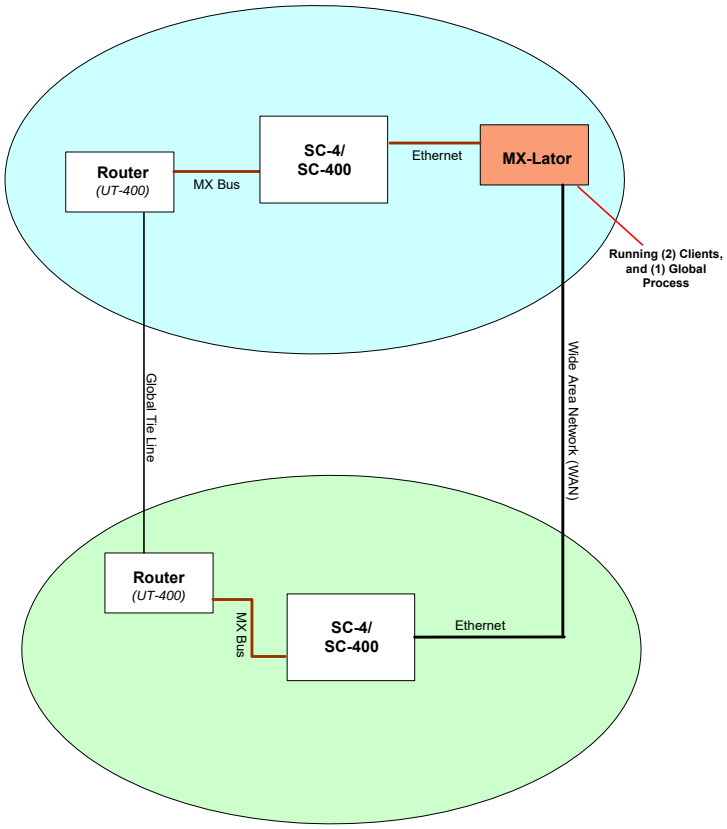


Figure 5-3. Two Locations - Option B

Global Process - One Geographic Location

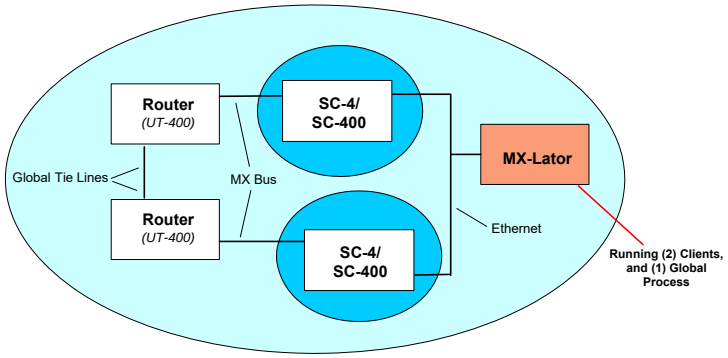


Figure 5-4. 1 Geographic Location

6

Chapter 6

‘MX-Lator’ Specifications

Chassis

- 2 RU chassis with the following features –
  - Dual Power Supplies
  - Dual Power Cord inlets
  - Two slots for SC-400 Control Cards
    - Loop Through MX Bus connectors
    - Two Serial Port Connectors
    - 12 Pin remote reset, changeover and 4 GPI/GPO connector
    - DB9 Alarm Connector.
    - Single UNET RJ45
    - Single 10/100 Ethernet RJ45.
  - Two slots for ‘MX-Lator’ Cards
    - 6 RJ45 Serial Ports for third party routers.
    - Dual SC-BUS DB9 connectors
    - Dual DB15 AVS1 Databus Connector.
    - Single 10/100 Ethernet RJ45.

MX-Lator Card Hardware Description

- 5307 / 5407 Coldfire MPU socket.
- 10/100 Ethernet MAC+PHY
- Two Quad Uarts for third party, SC-BUS and redundancy.
- MX Bus hardware Interface
- SC Bus hardware Interface
- FPGA for processor address decoding and custom MX-BUS Functionality.

A

Appendix A

Third Party Control Interfaces

Vendors and Protocols

Table 1-1. Master Control Automation Vendors

Harris (Louth)	
Harris (Drake)	
Sundance	
Florical	
NVerzion	
DTG	
RushWorks	

Table 1-2. Router Control Vendors

DNF Controls	Control and Status
Crestron	Control and Status
TSL	Status Display Only
Image Video	Status Display Only
Tally Display Corp.	Status Display Only
Leightronix	Control and Status
Howman	Status Display Only
RushWorks	Control and Status

Table 1-3. Production Switcher Vendors

Ross Video	Control and Status
Snell & Wilcox	Control and Status

Table 1-4. MX-Lator Control Protocols

			Refer to Note # Below
Thompson GVG	Jupiter	ES-Bus	3, 4, 5, 6
Thompson GVG	SMS-7000		13, 14
Probel			11, 12
Sandar	Prosan		10
Datatek			7, 8
Sierra Video			9
Pesa	3500		16

MX-Lator Protocol Details

1. UTAH DATA BUS, MASTER, PROPRIETARY BUS
   - 25 bit 320 X 320 X 7
   - 25 bit 320 X 160 X 8
   - 24 bit PL160
   - 24 bit PL150

Note: There are some limitations on 24-bit system compatibility. Check with Tech Support for details on specific systems.

2. UTAH SC BUS, MASTER, PROPRIETARY BUS
3. ESBUS CONTROLLER, MASTER, SERIAL

   (Note that all of our ES bus protocols do not do the multi-connection version.)

4. ESBUS TRIBUTARY, SLAVE, SERIAL

   (Note that all of our ES bus protocols do not do the multi-connection version.)

5. ESBUS TRIBUTARY CONFIRM, SLAVE, SERIAL

   (Note that all of our ES bus protocols do not do the multi-connection version.)

   This simply confirms all takes. This is used in situations were the third party controller cannot simulate for non-existing routers.

6. ESBUS CONTROLLER, BRIDGE, SERIAL

   (Note that all of our ES bus protocols do not do the multi-connection version.)

   Two types of bridges can be configured. One where the physical router is on the ES side and one were the physical router is on the UTAH side. Any combination can be configured as long as they are unique in the dest/level space on ES and MX.

   ESBC is ES Bus Controller protocol. It allows USI to bridge to ES physical routers.

ESBC LCL is both Bus Control protocol and ES Tributary protocol. This allows for bridges with USI physical router. This requires two serial ports. The issue requiring this is that

Path Finding will not output physical sources over Bus Control protocol.

Note: When using ESBC LCL the status on the USI panels will only be the USI status, not the path finding source. The ESBC will report path-finding source and the ESBCL will report USI router source. The ESBC source will be ignored when the on ESBCL mappings.

7. DATA TEK, MASTER, SERIAL
8. DATA TEK, BRIDGE, SERIAL
9. SIERRA, MASTER, SERIAL
10. SANDAR PROSAN, SLAVE, SERIAL
11. PROBEL, SLAVE, ETHERNET

This interface is optimized for use with HARMONIC NMX. It may not work with others.

- Only one level, level 0. The port number to level mapping is not implemented in NMX.
- Changes made on the SC4 are not sent to Probel (it crashes NMX).
- HARMONIC NMX only reads status at power up. USI may not know status then, so default source is sent. HARMONIC NMX will then compare and send what it wants it to be. The NMX flashes between red and green during this.
- The error UDP socket will never send anything. It simply reads and throws away everything.
- If problems occur between MXLator and SC4. The Probel connection is closed. This forces HARMONIC NMX to do a power up status sequence. No other way to tell HARMONIC NMX about problems.

- The Probel always has a source. It defaults to 0. USI may not have a source. The default source is used as status in this case.

12. PROBEL, MASTER, SERIAL

    One level per serial port.

13. GRASS VALLEY SMS-7000 NATIVE, SLAVE, ETHERNET

    This protocol has been revised so that it no longer require the GVG to simulate hardware. The GVG has a limitation that if simulation is on, all crosspoints are simulated. In addition to the standard MXlator limitations, the following functions have not been implemented:

    – Name modes (we only run indexed)
    – Chop
    – Protects/Locks
    – Salvos
    – Monitor controls
    – Tie lines
    – Date and Time
    – Machine assignment
    – GVG specific errors and status

    Other GVG Native slave protocol notes:

    1) The GVG machine appears to have a lot of controll in the way it can configure the Native protcol, some of them probably do not work.

    2) The status seems very slow on some of the GVG equipment.

    3) The GVG machine can be configured to request status on a regular basis. This will allow the system to work as a bridge. Takes made on the USI side are moved to the GVG side thru the status request.

14. GRASS VALLEY SMS-7000 NATIVE, MASTER, ETHERNET

15. HCP, MASTER, ETHERNET

Hardware Control Protocol. This is an in house protocol, allowing Ethernet transport of MX-Bus data.

16. PESA 3500 CONTROLLER, MASTER, SERIAL

    - The protocol document is CPU Link Protocol No. 1 (P1) - Document No.81-9062-0407-0 Revision B
    - Only two commands are implemented: Change and Status.

GENERAL PROTOCOL COMPATABILITY and IMPLEMENTATION ISSUES

- We only switch routers. No machine control, no locks/protects, no attributes, nothing is read back.
- Actual switching is delayed and may be switched in a different order than the SC4/400.
- We only perform commands on the protocol that we need to get the job done. There are usually other commands available. It is possible that different third party implementations use different sets of commands.
- Some of these protocols have only been tested on a non-native controller. At times we talk to a Brand X controller using Brand Y protocol.
- Some times a third party router will require an extra-cost upgrade to include a specific protocol. (BTS is this way)
- If a third party is going to control a USI router, the third party may have to simulate a router so that we can grab the status and switch. This sometimes removes any absolute confirm issues.

- Master and Slave:

  Master and Slave are terms that define who has the panels and who has the physical router. The Master has the panels, and the Slave has the physical router.

  Master and Slave terminology can also be used to define the protocol interaction. The Master will usually initiate messages on the connection, start any message interaction, etc. The Slave will usually perform commands without objection; make requests of Master that can be turned down. This primarily comes into play if we are connecting to a Brand X controller that is running Brand Y protocol. It is possible to be a Master as far as panels goes, but a slave in the protocol realm.

- Some of the protocols can run indexed sources/destinations into the SC4 (Datatek is one). If this is used, The RCP3/4 protocol must be enabled.

Index