

## TABLE OF CONTENTS

1.	OVERVIEW .....	1
2.	CARD EDGE CONTROLS.....	3
2.1.	DETERMINING CURRENT IP ADDRESS SETTINGS .....	3
2.2.	RESTORING FACTORY DEFAULTS.....	3
2.3.	CARD EDGE LEDS .....	3
3.	TECHNICAL SPECIFICATIONS.....	4
3.1.	DATA INPUT SERIAL PORT.....	4
3.2.	ELECTRICAL.....	4
3.3.	PHYSICAL .....	4
4.	CONFIGURATION .....	5
4.1.	CONFIGURATION STEPS .....	5
4.2.	DEBUG/MONITOR PORT CONNECTION.....	5
4.3.	MAIN MENU.....	7
4.4.	NETWORK CONFIGURATION.....	7
4.5.	SERIAL PORT SETUP .....	8
4.5.1.	Parameters .....	8
4.5.2.	Back Plate.....	9
4.5.3.	RS-232 Wiring.....	10
4.5.4.	RS-422 Wiring.....	11
4.6.	SNMP SETUP .....	12
4.7.	VM/SI 3000 ASCII PROTOCOL CONFIGURATION.....	12
4.7.1.	Number of Router Inputs and Outputs .....	12
4.7.2.	Input and Output Descriptions .....	13
4.7.3.	Level Usage .....	13
4.8.	UNDER MONITOR DISPLAY SETUP .....	14
5.	TROUBLESHOOTING TIPS.....	15
5.1.	VLPRO NOTES.....	15
5.2.	CHECKING CONTROLLER COMMUNICATION .....	15
5.3.	CHECKING UMD PEER COMMUNICATION .....	16
5.4.	CONTROLLER POLLING.....	16
6.	PERFORMING A FIRMWARE UPGRADE.....	18
6.1.	FTP PROCEDURE.....	18



6.2. SERIAL PROCEDURE ..... 18

7. VISTALINK® REMOTE MONITORING/CONTROL ..... 20

7.1. WHAT IS VISTALINK®?..... 20

Figures

Figure 1-1: PTX Block Diagram ..... 1

Figure 1-2: Typical 7700PTX-VMSI Setup ..... 2

Figure 2-1: PTX Card Edge ..... 3

Figure 4-1: Upgrade Jumper ..... 5

Figure 4-2: 'Connect To' Window ..... 6

Figure 4-3: COM1 Properties ..... 6

Figure 4-4: 7700PTX Back Plate ..... 9

Figure 4-5: RS-232 Pins ..... 10

Figure 4-6: RS-422 Pins ..... 11

Figure 4-7: UMD Example ..... 14

Figure 5-1: Communication States ..... 16

Tables

Table 4-1: 7700PTX-VMSI Main Menu..... 7

Table 4-2: Serial Port Parameters ..... 8

Table 4-3: RS-232 Wiring ..... 10

Table 4-4: RS-422 Wiring ..... 11

Table 4-5: SNMP Parameters ..... 12

## REVISION HISTORY

<u>REVISION</u>	<u>DESCRIPTION</u>	<u>DATE</u>
1.0	Preliminary	Dec 05
1.1	Standardized Format	Mar 07
1.2	Updated card edge drawing	Nov 07
1.3	Added features, block diagram, technical specs & <i>VistaLINK</i> ® section.	Nov 08
1.4	Removed references to GPI, GPO, LTC specifications	Apr 09

Information contained in this manual is believed to be accurate and reliable. However, Evertz assumes no responsibility for the use thereof nor for the rights of third parties, which may be affected in any way by the use thereof. Any representations in this document concerning performance of Evertz products are for informational use only and are not warranties of future performance, either expressed or implied. The only warranty offered by Evertz in relation to this product is the Evertz standard limited warranty, stated in the sales contract or order confirmation form.

Although every attempt has been made to accurately describe the features, installation and operation of this product in this manual, no warranty is granted nor liability assumed in relation to any errors or omissions unless specifically undertaken in the Evertz sales contract or order confirmation. Information contained in this manual is periodically updated and changes will be incorporated into subsequent editions. If you encounter an error, please notify Evertz Customer Service department. Evertz reserves the right, without notice or liability, to make changes in equipment design or specifications.

*This page left intentionally blank*

## 1. OVERVIEW

The 7700PTX Universal Protocol Translator module provides an interface between third-party and Evertz equipment. The 7700PTX communicates with third-party equipment either via one of four serial ports or via a built in Ethernet port. These ports can provide bi-directional protocol support.

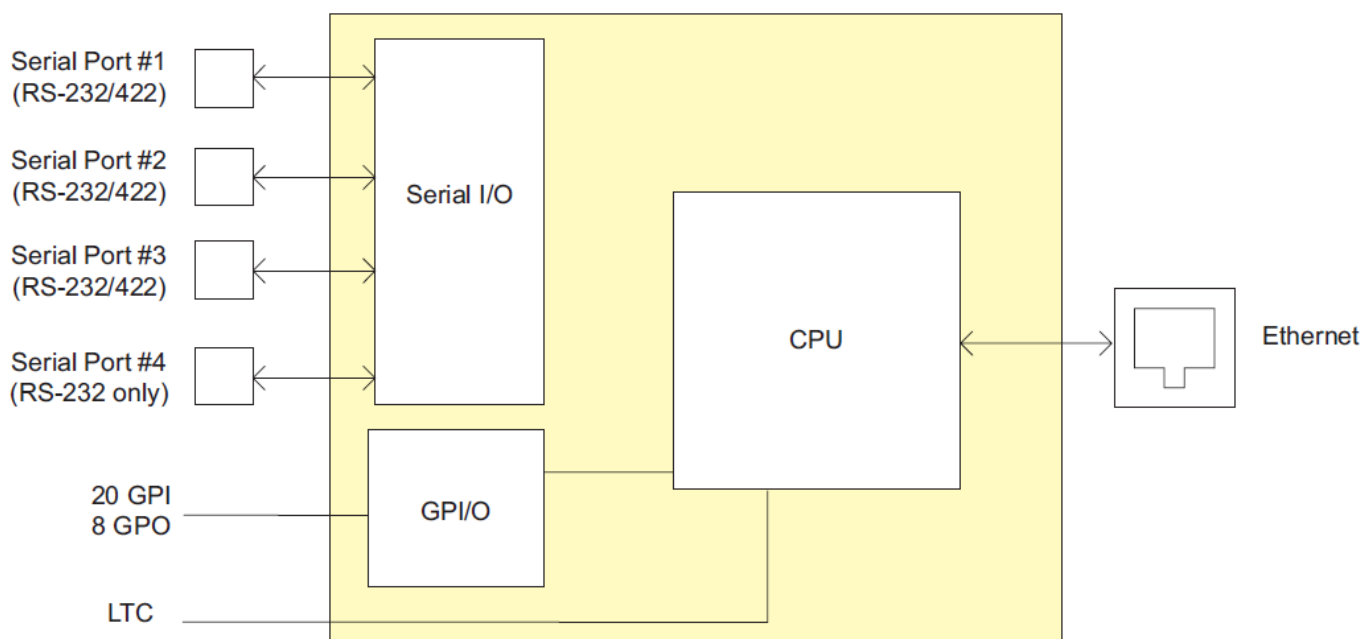
### Function:

The function of the 7700PTX generally falls into one of 4 categories:

1. **Third-Party Router Control:** In this mode the 7700PTX affords *VistaLINK*® the ability to control and monitor third-party routers. The 7700PTX can convey UMD information to Evertz monitoring equipment.
2. **Third-Party UMD Interface:** In this mode the 7700PTX translates third-party UMD protocol data into a format suitable for Evertz monitoring equipment.
3. **Third-Party Switcher Interface:** In this mode the 7700PTX extracts tally information from third-party switchers and translates and conveys that tally information to Evertz monitoring equipment.
4. **Third-Party Device Control:** In this mode the 7700PTX allows *VistaLINK*® to control third-party devices such as satellite controllers.

### Features:

- 4 serial ports RS232/422 selectable
- Modular, conveniently fits into 7700FR-C 3RU frame
- Frame status trigger
- *VistaLINK*® - capable for remote monitoring and control via SNMP (using *VistaLINK*®PRO)



**Figure 1-1: PTX Block Diagram**

The 7700PTX-VMSI is designed to control and/or monitor a controller via a Jupiter controller. The controller employs the VM/SI 3000 ASCII protocol. The 7700PTX-VMSI can operate in either or both of two modes:

1. **A control mode:** *VistaLINK*<sup>®</sup> Pro performs controller cross-point switches.
2. **A monitor mode:** The 7700PTX-VMSI monitors the controller for cross point changes and transmits label or description information to a UMD (Under Monitor Display).

The 7700PTX-VMSI is a network-controlled protocol translator that translates SNMP (Simple Network Management Protocol) application commands into VM/SI 3000 protocol packets. The translated packets are then transmitted to one of up to four Jupiter controllers. These controllers are connected serially to the 7700PTX-VMSI.

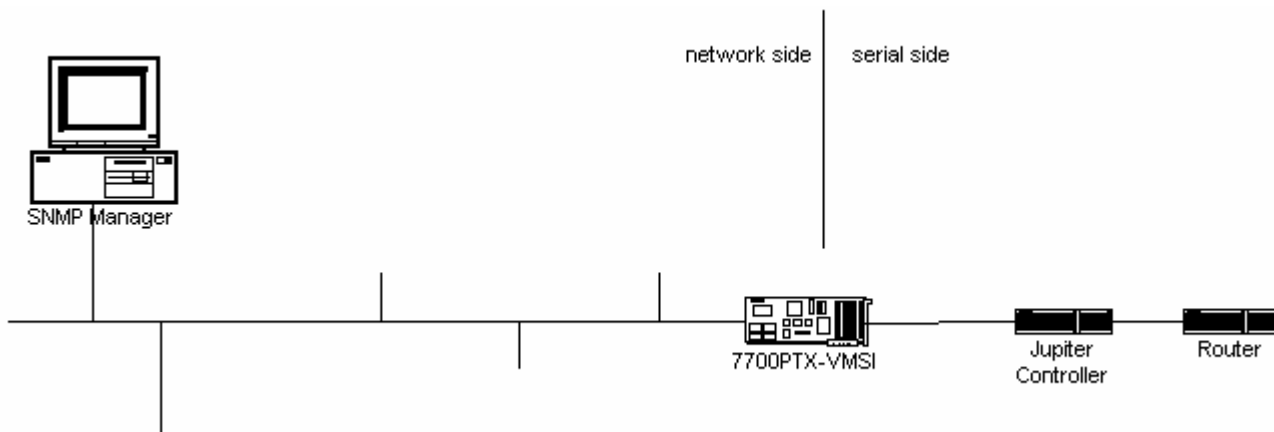
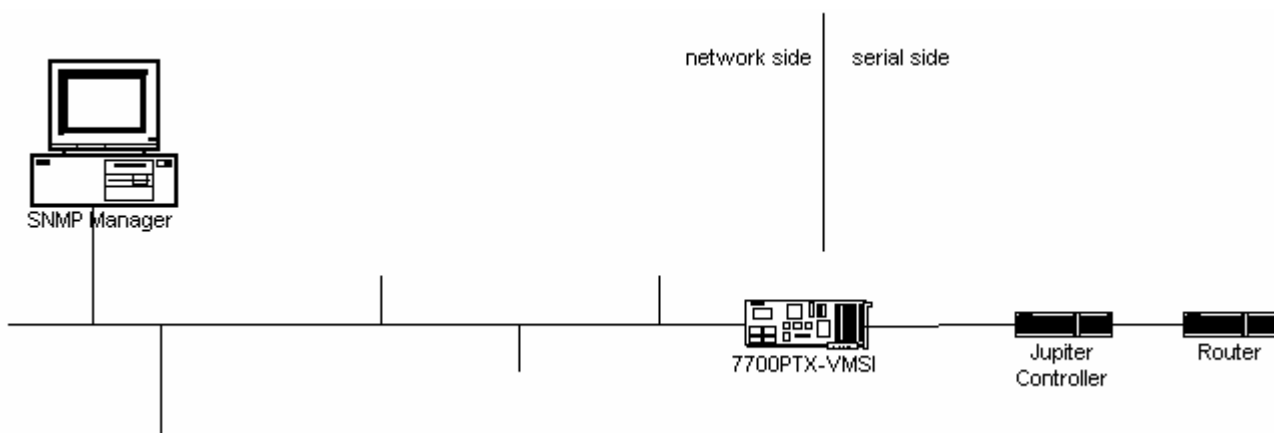


Figure 1-2 shows a typical 7700PTX-VMSI setup.



**Figure 1-2: Typical 7700PTX-VMSI Setup**

## 2. CARD EDGE CONTROLS

### 2.1. DETERMINING CURRENT IP ADDRESS SETTINGS

To read the current IP address during normal operation, press the toggle switch DOWN. The IP address can be read on the four-character alphanumeric display.

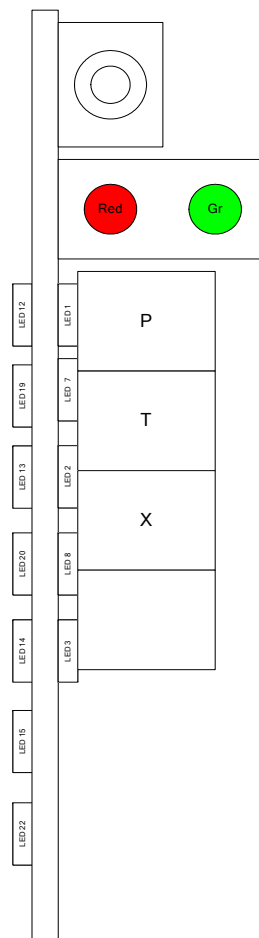
### 2.2. RESTORING FACTORY DEFAULTS

To restore all settings to factory defaults, apply power to the card while holding the toggle switch UP until the green LED is illuminated.

### 2.3. CARD EDGE LEDS

LED 22 is illuminated when Ethernet activity is detected.

All other card edge LEDs are for factory use only.



**Figure 2-1: PTX Card Edge**

### **3. TECHNICAL SPECIFICATIONS**

#### **3.1. DATA INPUT SERIAL PORT**

**Number of Ports:** 4 RS-232 or 3 RS-422  
**Connector:** Phoenix Terminal Block pins  
**Baud Rate:** Up to 1Mbaud

#### **3.2. ELECTRICAL**

**Voltage:** +12V DC  
**Power:** < 6W  
**Safety:** ETL Listed, complies with EU safety directives  
**EMI/RFI:** Complies with FCC Part 15, Class A  
EU EMC Directive

#### **3.3. PHYSICAL**

**Number of slots:** 2

## **4. CONFIGURATION**

### **4.1. CONFIGURATION STEPS**

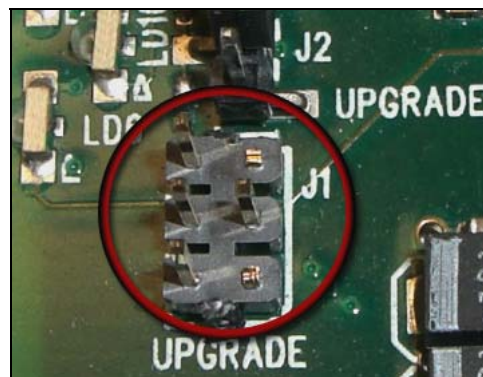
Perform the following steps to configure the 7700PTX-VMSI:

1. Connect a PC running a console application to the PTX debug/monitor port via the adapter cable.
2. Configure the 7700PTX-VMSI's network parameters.
3. Configure the parameters of each serial port to match those of the Jupiter controller(s).
4. Configure the parameters associated with the VM/SI 3000 ASCII protocol.
5. Configure UMD peers if required.
6. Power off the 7700PTX-VMSI.
7. Physically wire the serial port(s) of the 7700PTX-VMSI to the Jupiter controller(s).
8. Power on the 7700PTX-VMSI.

### **4.2. DEBUG/MONITOR PORT CONNECTION**

The 7700PTX-VMSI is configured via the debug/monitor port, the header of which is labelled J1. A special Evertz adapter cable allows this port to connect to the COM port of a personal computer. The following steps describe this procedure.

1. Locate the small, keyed, four-pin end of the upgrade cable provided by Evertz.
2. Connect it to the four-pin interface (J1) near the front of the 7700PTX, directly above the card unlock latch.



**Figure 4-1: Upgrade Jumper**

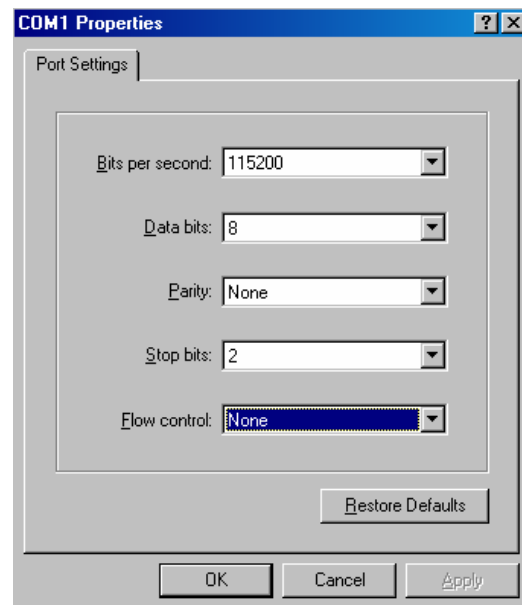
3. Connect the other end of the upgrade cable to a straight-through serial cable. Connect the serial cable to the serial or COM port of the computer.

4. Initiate HyperTerminal on your computer by selecting:  
“Start\Programs\Accessories\Communications\HyperTerminal”.
5. Enter a name for your connection, for example: PTX.
6. Press the <Enter> key. A new “Connect To” window opens.



**Figure 4-2: ‘Connect To’ Window**

7. Select COM1 for the “*Connect using*” setting. If COM1 is in use, choose an alternate COM port.
8. Press the <Enter> key or select OK. This opens the “COM Properties” window as shown in Figure 4-3.



**Figure 4-3: COM1 Properties**

9. Enter the information as listed in Figure 4-3.
10. Press the <Enter> key or select OK. The “COM Properties” window closes, leaving the *HyperTerminal* window open.
11. Apply power if the 7700PTX- VMSI does not have power. The boot sequence and Main Menu are displayed in the HyperTerminal window.
12. If the 7700PTX-VMSI has power, press the <Enter> key to view the 7700PTX- VMSI's menu system.
13. Various 7700PTX- VMSI parameters are configurable via the 7700PTX- VMSI's menu system, the root of which is called *Main Menu*.

### 4.3. MAIN MENU

Table 4-1 lists the entries available in the 7700PTX-VMSI's *Main Menu*.

Entry	Item	Notes
1	<b>Network Configuration</b>	IP address, subnet mask, gateway, etc.
2	<b>Serial Port Setup</b>	Baud rate, number of data bits, etc. of serial ports which connect to Jupiter controller(s)
3	<b>SNMP Setup</b>	IP address of SNMP manager(s) to receive traps
4	<b>VM/SI 300 ASCII Protocol Settings Setup</b>	Settings specific to the controller and the VM/SI 3000 ASCII protocol.
5	<b>Under Monitor Display Setup</b>	IP address and TCP port of PPV to receive the description of the input associated with a particular output
6	<b>Engineering/Debug</b>	Used for troubleshooting

**Table 4-1: 7700PTX-VMSI Main Menu**

### 4.4. NETWORK CONFIGURATION

1. From the *Main Menu* select *Network Configuration*.
2. If DHCP (Dynamic Host Configuration Protocol) is desired, then the *Use DHCP* field is set to *True*. Otherwise, the IP address, subnet mask, and gateway (if any) are set and the *Use DHCP* field is set to *False*.
3. Once the network settings are configured, select *Save* and *Exit* before exiting *Network Configuration* to save the settings, otherwise select *Exit*.



**The 7700PTX-VMSI must be rebooted for any network setting changes to take effect.**

## **4.5. SERIAL PORT SETUP**

### **4.5.1. Parameters**

The 7700PTX-VMSI has 4 serial ports. The parameters associated with each serial port are list in Table 4-2.

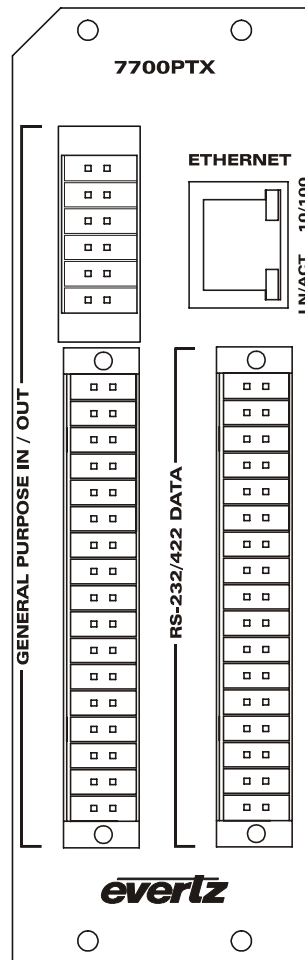
Parameter	Special Notes
Baud Rate	
Data Bits	
Parity	
Stop Bits	
Standard	For serial port 4, only RS-232 is valid.

**Table 4-2: Serial Port Parameters**



The serial port settings of the 7700PTX-VMSI must match those of the controller(s). The 7700PTX-VMSI must be rebooted for any serial parameter changes to take effect.

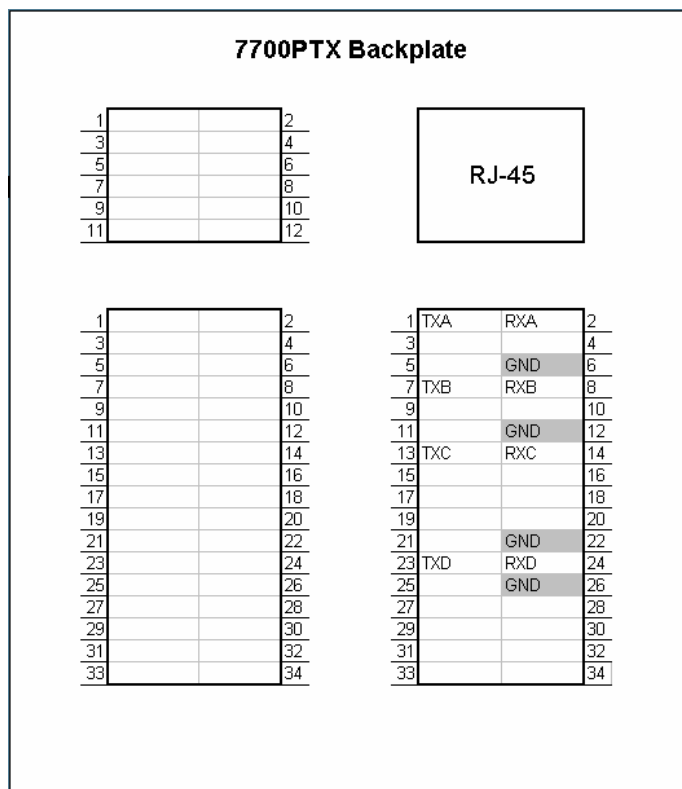
### 4.5.2. Back Plate



**Figure 4-4: 7700PTX Back Plate**

### 4.5.3. RS-232 Wiring

Figure 4-5 shows which pins of the back plate are used for RS-232 serial connections.



**Figure 4-5: RS-232 Pins**

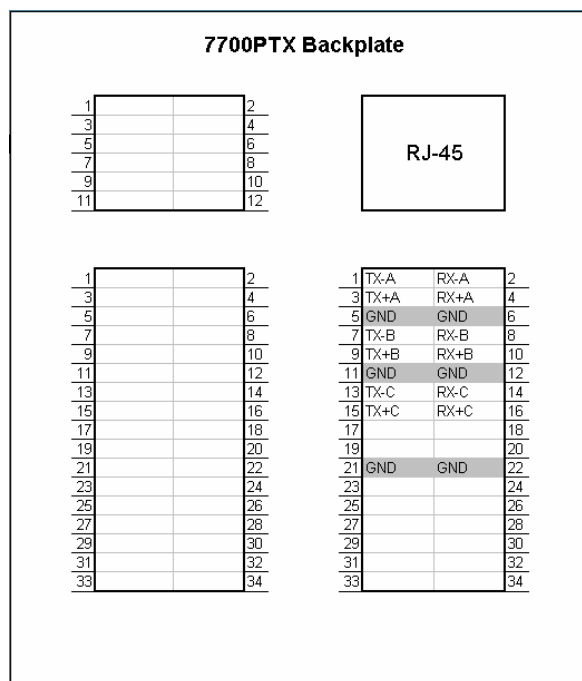
Table 4-3 details how to connect the 7700PTX-VMSI to the controller for RS-232 operation.

7700PTX-VMSI			Controller
Port	Pin Name	Pin	Pin Name
1	TXA	1	RX
	RXA	2	TX
	GND	6	GND
2	TXB	7	RX
	RXB	8	TX
	GND	12	GND
3	TXC	13	RX
	RXC	14	TX
	GND	22	GND
4	TXD	23	RX
	RXD	24	TX
	GND	26	GND

**Table 4-3: RS-232 Wiring**

#### 4.5.4. RS-422 Wiring

Figure 4-6 shows which pins of the back plate are used for RS-422 serial connections.



**Figure 4-6: RS-422 Pins**

Table 4-4 details how to connect the 7700PTX-VMSI to the controller for RS-422 operation.

7700PTX-VMSI			Controller
Port	Pin Name	Pin	Pin Name
1	TX-A	1	RX-
	TX+A	3	RX+
	RX-A	2	TX-
	RX+A	4	TX+
	GND	6	GND
2	TX-B	7	RX-
	TX+B	9	RX+
	RX-B	8	TX-
	RX+B	10	TX+
	GND	12	GND
3	TX-C	13	RX-
	TX+C	15	RX+
	RX-C	14	TX-
	RX+C	16	TX+
	GND	22	GND

**Table 4-4: RS-422 Wiring**



The 7700PTX-VMSI's 4th serial port is not RS-422 capable.

## 4.6. SNMP SETUP

Table 4-5 lists the parameters associated with the SNMP setup.

Parameter	Notes
Read-only community	Community string used for SNMP gets. The default is <i>public</i> .
Read-write community	Community string used for SNMP gets or sets. The default is <i>private</i> .

**Table 4-5: SNMP Parameters**



**These parameters must match those of the SNMP manager. Changes to these parameters do not require a reboot of the 7700PTX-VMSI.**

## 4.7. VM/SI 3000 ASCII PROTOCOL CONFIGURATION



**Changes to any of these parameters do not require a reboot of the 7700PTX-VMSI.**

### 4.7.1. Number of Router Inputs and Outputs

The VM/SI 3000 ASCII protocol does not permit the 7700PTX-VMSI to automatically determine the size of the router. Thus, the number of router inputs and outputs must be set on the 7700PTX-VMSI. This can be accomplished by:

- Using *VistaLINK*<sup>®</sup> Pro
- Using the 7700PTX-VMSI's *VM/SI 300 ASCII Protocol Settings Setup* menu

To configure the number of router inputs and outputs from the console follow the steps below:

1. From the *Main Menu* select *VM/SI 3000 ASCII Protocol Settings Setup*.
2. Select *VM/SI Protocol Setup For Serial Port 1* (or the relevant serial port to which the Jupiter controller is connected).
3. Select *Set number of router inputs*.
4. Enter the number of router inputs.
5. Select *Set number of router outputs*.
6. Enter the number of router outputs.
7. Select *Save and Exit*.

#### **4.7.2. Input and Output Descriptions**

The VM/SI 3000 ASCII protocol does not permit the 7700PTX-VMSI to automatically determine the input or output labels/descriptions. Thus, the descriptions must be set on the 7700PTX-VMSI. This can be accomplished by:

- Using VistaLink® Pro
- Using the 7700PTX-VMSI's *VM/SI 300 ASCII Protocol Settings Setup* menu

To configure the input descriptions from the console follow the steps below:

1. From the *Main Menu* select *VM/SI 3000 ASCII Protocol Settings Setup*.
2. Select *VM/SI Protocol Setup For Serial Port 1* (or the relevant serial port to which the Jupiter controller is connected).
3. Select *Set input description* or *Set output description*.
4. Enter the router input or output.
5. Enter the description.
6. Repeat the process for all router descriptions.
7. Select *Save and Exit*.

#### **4.7.3. Level Usage**

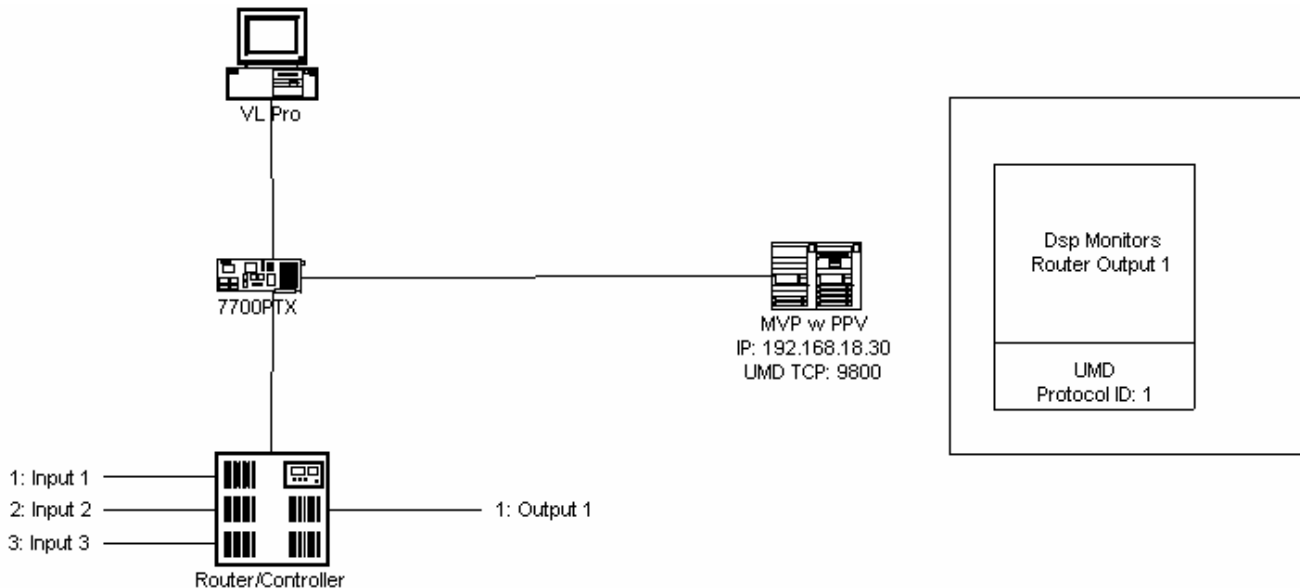
Up to seven levels are supported by the VM/SI 3000 ASCII Protocol. By default the 7700PTX-VMSI monitors level 1 and omits the level field in its switch request implying all defined levels are to be switched. If different levels are to be monitored/switched follow the steps below:

1. From the *Main Menu* select *VM/SI 3000 ASCII Protocol Settings Setup*.
2. Select *VM/SI Protocol Setup For Serial Port 1* (or the relevant serial port to which the Jupiter controller is connected).
3. Select *Enable/disable level usage*.
4. Confirm level usage by entering *y*.
5. Enter the level field.
6. Select *Save and Exit*.

#### 4.8. UNDER MONITOR DISPLAY SETUP

The 7700PTX-VMSI has the ability to transmit controller source label information to the UMDs of up to 12 PPVs.

As an example, suppose we have the setup of Figure 4-7.



**Figure 4-7: UMD Example**

##### Where:

- A controller has 3 inputs connected (labeled Input 1, Input 2, and Input 3) and 1 output (labeled Output 1)
- A 7700PTX-VMSI monitors the controller cross points
- An MVP contains a PPV with IP address 192.168.18.30
- The PPV is set to receive UMD data via the Image Video protocol over a TCP, with TCP port configured at 9800
- Protocol ID (PID) set to 1
- A PC running VistaLINK<sup>®</sup> Pro configures the 7700PTX-VMSI so that the UMD PID associated with controller Output 1 matches the PID of the UMD (for example, 1)

The *Under Monitor Display Setup* menu allows the configuration of the IP address and TCP port of the PPV to receive controller source label information. In keeping with the above example, the 7700PTX-VMSI would be configured to have a peer 1 IP address of 192.168.18.30 and a TCP port of 9800. When controller input 1 is on output 1, the UMD of the display should display INPUT 1. If the cross point is switched to input 3, the UMD should display INPUT 3.



**The 7700PTX-VMSI must be rebooted for any UMD peer changes to take effect.**

## **5. TROUBLESHOOTING TIPS**

### **5.1. VLPRO NOTES**

1. The 7700PTX-VMSI must be able to communicate with any connected controllers in order for VLPro to operate properly.
2. The 7700PTX-VMSI must be able to communicate with its configured UMD peers before UMD information can be transmitted.
3. VLPro must associate a UMD protocol ID with a controller output in order for UMD information to be transmitted.

### **5.2. CHECKING CONTROLLER COMMUNICATION**

1. From the *Main Menu* select *Engineering/Debug*.
2. Select *Show task state*.
3. There are four entries, one for each serial port, list under the heading *Controller protocol PCB state...* If the state associated with the serial port is reported as *ready* then the 7700PTX-VMSI is actively communicating with the controller on that port. If the state is consistently reported as *down* then the 7700PTX-VMSI is unable to communicate with the controller in which case the serial port settings or wiring should be checked.
4. Figure 5-1 shows the 7700PTX-VMSI is able to communicate with a controller connected to port 2.

```

dbg_1_115 - HyperTerminal
File Edit View Call Transfer Help

UMD peer status...
Port      Dst Address      Tcp Port      Status
****      *****
E1        192.168.18.40    9800         ready
E1        0.0.0.0          0            no address set
E1        0.0.0.0          0            no address set
E1        0.0.0.0          0            no address set
E1        0.0.0.0          0            no address set
E1        0.0.0.0          0            no address set
E1        0.0.0.0          0            no address set
E1        0.0.0.0          0            no address set
E1        0.0.0.0          0            no address set
E1        0.0.0.0          0            no address set
E1        0.0.0.0          0            no address set
E1        0.0.0.0          0            no address set

router protocol task state: ready

router protocol PCB state...
1        down
2        ready
3        down
4        down

Connected 2:36:23  ANSI  115200 8-N-2  SCROLL  CAPS  NUM  Capture  Print echo
  
```

**Figure 5-1: Communication States**

### 5.3. CHECKING UMD PEER COMMUNICATION

1. From the *Main Menu* select *Engineering/Debug*.
2. Select *Show task state*.
3. The status of the 12 UMD peer entries are listed under the heading *UMD peer status...* A status reported as *ready* indicates the 7700PTX-VMSI is able to communicate with that UMD peer. A status consistently reported as something other than *ready* indicates the inability of the 7700PTX-VMSI to communicate with that UMD peer. Be sure that the UMD peer has been rebooted after being configured to receive the Image Video over TCP.
4. Figure 5-1 shows the 7700PTX-VMSI able to communicate with the UMD peer whose IP address is 192.168.18.40 and who is listening on TCP port 9800.

### 5.4. CONTROLLER POLLING

By default, the 7700PTX-VMSI sends a *watch all defined outputs* request message every three minutes. This polling message serves two purposes:

1. To refresh the outputs watched
2. To monitor the connection status between the 7700PTX-VMSI and the Jupiter controller.

This poll duration can be changed via the *Set Router Poll Status* entry of the *Engineering/Debug* menu. The time between polls can be set on a per-serial port basis. If 0 is selected as the time between polls, polling is disabled.

Once set, the router poll status setting takes effect immediately; no reboot is necessary. It is saved to flash and recalled should the 7700PTX-VMSI be powered off, and then on.

## **6. PERFORMING A FIRMWARE UPGRADE**

There are two ways to upgrade PTX firmware:

1. Using FTP to perform the upgrade via TCP/IP. (*recommended procedure*)
2. Using a terminal application such as *HyperTerminal* to perform the upgrade via a serial connection.

### **6.1. FTP PROCEDURE**

1. Open a command prompt window (in Windows: Start/Programs/Accessories/Command Prompt)
2. Enter the location of the firmware file. For example, type `cd c:\temp`.
3. Enter the command `ftp` followed by the PTX IP address.  
For example, type `ftp -A 192.168.18.22`.
4. Enter the FTP command `put` followed by the firmware file name. For example, `put ptx.bin`.
5. When the transfer is complete enter the FTP command: `bye`.
6. Step 5 begins the process of saving the firmware to the non-volatile flash of the PTX. The save process is displayed as a percentage on the PTX LCD. Once the process is complete, the PTX LCD again displays the product name and firmware version.
7. Power off the PTX.
8. Power on the PTX.

### **6.2. SERIAL PROCEDURE**

1. Power off the PTX.
2. Connect an adapter cable to a PC running a console or terminal application, such as Windows *HyperTerminal*, to the PTX debug/monitor port.
3. Configure the port settings of the terminal program as follows:

Baud	<b>115200</b>
Parity	<b>no</b>
Data bits	<b>8</b>
Stop bits	<b>2</b>
Flow Control	<b>None</b>

4. Set the PTX run/upgrade jumper to the upgrade position.
5. Power on the PTX.
6. After a few moments, the prompt `PPCBOOT>` will appear. Enter the command `upload`.

7. Start the firmware upload on the terminal application (for instance, in *HyperTerminal* select Transfer/Send File...), use Xmodem as the transfer protocol, and select the firmware file. For example, *ptx.bin*.
8. Once the upload is complete the message *upload okay* is displayed.
9. Power off the PTX.
10. Set the PTX run/upgrade jumper to the run position.
11. Remove the serial adapter cable.
12. Power on the PTX.

## **7. VISTALINK® REMOTE MONITORING/CONTROL**

### **7.1. WHAT IS VISTA LINK®?**

VistaLINK® is Evertz' remote monitoring and configuration platform which operates over an Ethernet network using Simple Network Management Protocol (SNMP). SNMP is a standard computer network protocol that enables different devices sharing the same network to communicate with each other. VistaLINK® provides centralized alarm management, which monitors, reports, and logs all incoming alarm events and dispatches alerts to all the VLPro Clients connected to the server. Card configuration through VistaLINK® PRO can be performed on an individual or multi-card basis using simple copy and paste routines, which reduces the time to configure each module separately. Finally, VistaLINK® enables the user to configure devices in the network from a central station and receive feedback that the configuration has been carried out.

There are 3 components of SNMP:

1. An SNMP manager, also known as a Network Management System (NMS), is a computer running special software that communicates with the devices in the network. Evertz VistaLINK®-C Configuration Utility graphical user interface (GUI), third party or custom manager software may be used to monitor and control Evertz VistaLINK® enabled products.
2. Managed devices, (such as 7700PTX modules), each with a unique address (OID), communicate with the NMS through an SNMP Agent. The 7700PTX-VMSI communicates directly with the manager using its internal Agent.
3. A virtual database known as the Management information Base (MIB) lists all the variables being monitored, which both the Manager and Agent understand. Please contact Evertz for further information about obtaining a copy of the MIB for interfacing to a third party Manager/NMS.