



LeMans
VX8 Carrier Board
Installation Guide

Document Number 500-00328

Revision 2.01

September 1998

Copyright © 1998 Spectrum Signal Processing Inc.

All rights reserved, including those to reproduce this document or parts thereof in any form without permission in writing from Spectrum Signal Processing Inc.

All trademarks are registered trademarks of their respective owners.

Spectrum Signal Processing reserves the right to change any of the information contained herein without notice.

Customer Feedback

At Spectrum, we recognize that product documentation that is both accurate and easy to use is important in aiding you in your new product development. We appreciate hearing your comments on how our product's documentation could be improved.

If you wish to comment on any Spectrum documentation then please fax or e-mail a completed copy of this page to us.

Full Name of Document: _____

Document Number: _____ Version Number: _____

If you have found a technical inaccuracy please describe it here:

If you particularly liked or disliked an aspect of the manual then please describe it here:

It may be helpful for us to call you to discuss your comments. If this would be acceptable please provide the following details:

Name: _____ Telephone #: _____

Organization: _____

Thank you for your time,

Spectrum Signal Processing Documentation Group

Fax: (604) 421-1764

Email: documentation@spectrumsignal.com

Contacting Spectrum...

Spectrum's team of dedicated Applications Engineers are available to provide technical support to you for this product. Our office hours are Monday to Friday, 8:00 AM to 5:00 PM, Pacific Standard Time.

Telephone 1-800-663-8986 or (604) 421-5422

Fax (604) 421-1764

Email support@spectrumsignal.com

Internet <http://www.spectrumsignal.com>

When you contact us, please have the following information on hand:

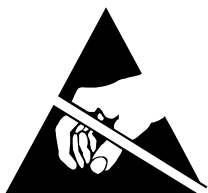
- A concise description of the problem
- The name of all Spectrum hardware components
- The name and version number of all Spectrum software components
- The minimum amount of code that demonstrates the problem
- The version number of all software packages, including compilers and operating systems

Preface

Spectrum Signal Processing offers a complete line of DSP hardware, software and I/O products for the DSP Systems market based on the latest DSP microprocessors, bus interface standards, I/O standards and software development environments. By delivering quality products, and DSP expertise tailored to specific application requirements, Spectrum can consistently exceed the expectations of our customers. We pride ourselves in providing unrivaled pre and post sales support from our team of application engineers. Spectrum has excellent relationships with third party vendors which allows us to provide our customers with a more diverse and top quality product offering.



Spectrum achieved ISO 9001 quality certification in 1994.



As Spectrum's hardware products are static sensitive, please take precautions when handling and make sure they are protected against static discharge.

Table of Contents

1 Introduction	1
1.1. Purpose of This Manual.....	1
1.2. General Overview.....	1
1.2.1. Hardware Overview	1
1.2.2. Software Overview.....	4
2 Hardware Installation and Configuration	5
2.1. Removing the VX8 from a VXI Mainframe.....	6
2.2. Removing the Top Shield	7
2.3. Installing TIM-40 Modules	9
2.4. DRAM Memory Requirements.....	10
2.5. Installing DRAM SIMMs.....	11
2.6. Jumpers and DIP Switches	13
2.6.1. Setting the VXIbus A16 Logical Address.....	14
2.6.2. Selecting the Node B Boot Mode.....	15
2.6.3. Setting the Node A Kernel PEROM Write Enable	16
2.6.4. DRAM Memory Installation.....	16
2.6.5. TIM-40 TCLK1/UDP12 Sources	17
2.7. Replacing the Top Shield	18
2.8. Installing the VX8 in a VXI Mainframe	19
3 VX8 C4x Support Software Installation	21
3.1. Directory Structure.....	21
3.2. Requirements	22
3.3. Installing the VX8 C4x Support Software	22
3.4. Software Configuration Notes.....	23
3.5. Using the Example Code.....	25
4 VX8 Instrument Driver Installation.....	27
4.1. HP-UX 9.0x SICL Instrument Driver Installation.....	27
4.1.1. Tape Contents	27
4.1.2. Requirements	28
4.1.3. Installing the VX8 SICL Instrument Driver from Tape.....	29
4.2. Windows 95/WinNT VISA Instrument Driver Installation.....	30
4.2.1. Installation Description.....	30

4.2.2. Requirements	31
4.2.3. Installing the VX8 VISA Instrument Driver from Diskette.....	32
4.2.4. Software Configuration Notes.....	32
4.2.5. Hardware Configuration Notes	33
5 System Definition File Setup	35

List of Figures

Figure 1 VX8 Carrier Board Configuration	2
Figure 2 VX8 Front Panel.....	3
Figure 3 VX8 Jumper and Switch Locations	13
Figure 4 A16 Address DIP Switch.....	14
Figure 5 DRAM Jumper Settings	17
Figure 6 CLK10, ECLTRG0 and ECLTRG1 Jumpers.....	18

List of Tables

Table 1 DRAM Sizes and Jumper Settings.....	11
Table 2 Boot Source Jumpers (JP1 and JP4).....	13
Table 3 DIP Switch A16 Addresses	15
Table 4 VX8 Developer's Kit Disk Contents.....	21
Table 5 VX8 Developer's Kit Tape Fileset.....	27
Table 6 VX8 Developer's Kit Tape Installed Directory Structure.....	28
Table 7 VX8 Instrument Driver VXiPnp Installed Files.....	30
Table 8 VX8 Instrument Driver Source Directory Structure	30

1 Introduction

1.1. Purpose of This Manual

This manual provides the information you need to install and configure the VX8 Carrier Board and Support Software. It provides a brief description of the VX8 hardware and software, and it details the steps required to install the VX8 Carrier Board, C4x Support Software, and the VX8 Instrument Driver. You'll also want to refer to the following two manuals when developing applications using the VX8 Carrier Board:

- *The VX8 Carrier Board Technical Reference Manual (TRM)*
- *The VX8 Carrier Board Programming Guide*

1.2. General Overview

The VX8 Carrier Board is part of Spectrum's series of VXIbus boards based upon Texas Instrument's TMS320C4x DSPs. The VX8 Carrier Board delivers high performance signal processing based on a modular DSP architecture that allows you to create powerful DSP-based systems for the VXI environment.

1.2.1. Hardware Overview

The VX8 Carrier Board is a VXIbus Master/Slave DSP board catered to high performance applications with large volumes of data throughput.

Base Configuration

The base configuration of the VX8 features:

- two embedded 60 MHz TMS320C40 DSPs (for a base subsystem of 120 MFLOPs),
- 1 MB of SRAM per C40 (512 kBytes per local and global buses),
- a VXIbus interface,
- front panel connections for C4x Communication Ports,
- an asynchronous dual UART (RS-232),
- and a JTAG front panel interface.

Optional Features

Optional features for the VX8 include:

- upgrades to shared DRAM,

- upgrades to embedded C40 local and global bus SRAM,
- the addition of TIM-40 modules,
- and the addition of a high-throughput HP Local Bus interface.

By populating the VX8 with up to six TIM-40 modules, the processing power can be increased to a maximum of 840 MFLOPS, providing high performance parallel computing for demanding applications. Adding the HP Local Bus provides efficient data transfer compatible with a wide range of Hewlett-Packard VXI I/O modules.

The backbone of the VX8 is the Global Shared Bus which interconnects the VXI interface and the HP Local bus with the shared DRAM and the global SRAM of the embedded C40's or the TIM-40 modules. The VXI interface can access C40/TIM global SRAM and the shared DRAM. The HP Local Bus can only access the C40/TIM global SRAM. C40/TIM SRAM can be accessed from other C40s/TIMs using the shared Global Bus but it is recommended to make use of the dedicated point-to-point communication ports and use the shared global bus for data transfers on and off the VX8. *Figure 1* illustrates the configuration of a fully-loaded VX8 Carrier Board. The unshaded areas represent the VX8 Carrier Board's base configuration.

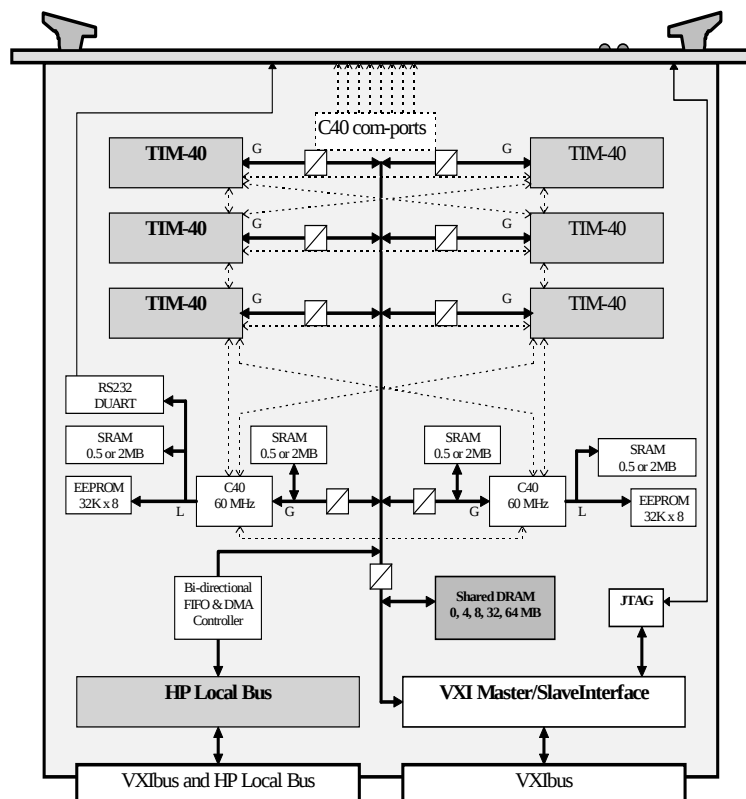


Figure 1 VX8 Carrier Board Configuration

Front Panel The front panel of the VX8 has a variety of connectors and status LEDs as shown in the following illustration. Refer to the *VX8 Carrier Board Technical Reference Manual* for details.

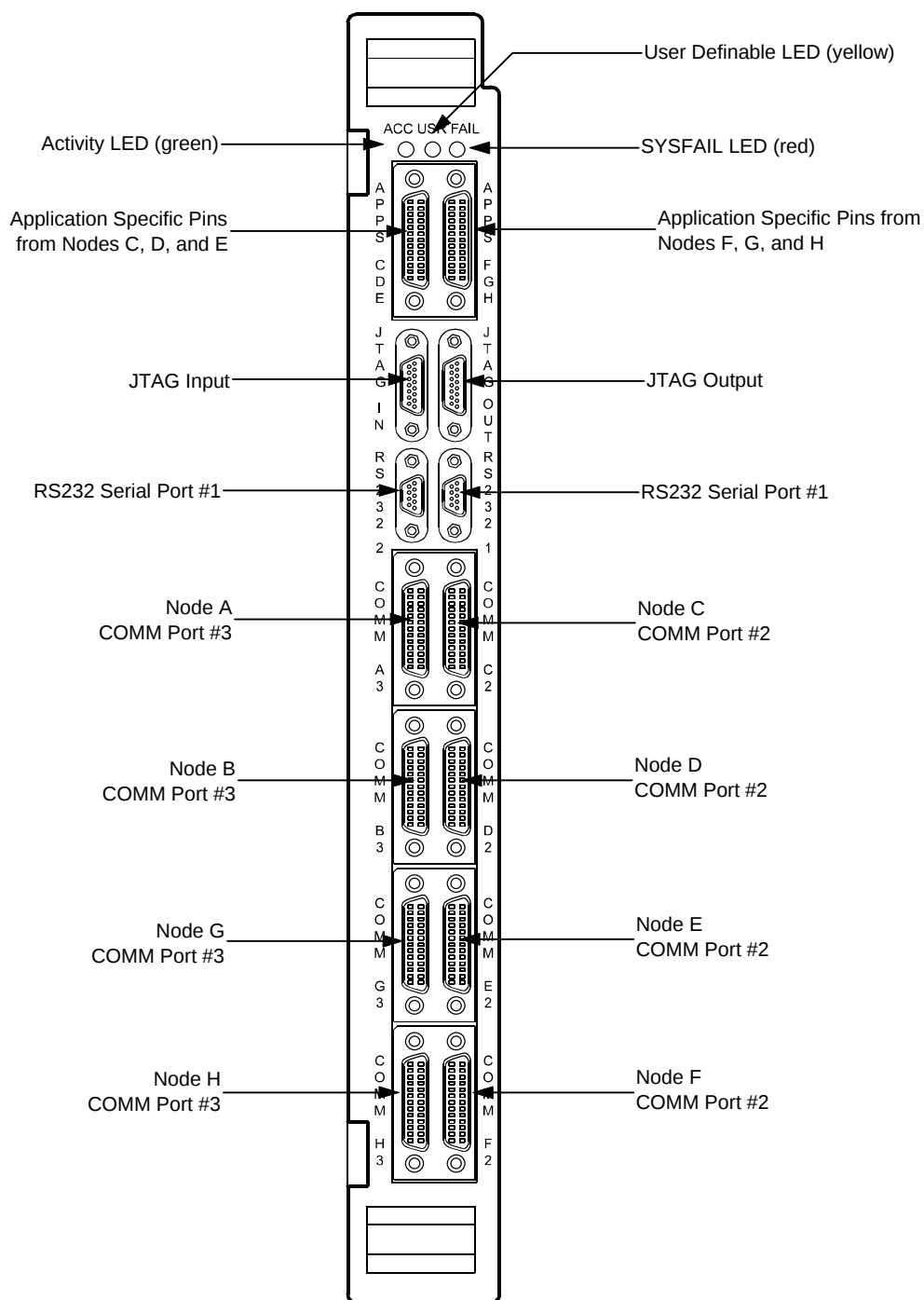


Figure 2 VX8 Front Panel

1.2.2. Software Overview

The VX8 Support Software offers HP-UX 9.05 SICL support. WinNT VISA will be supported in future releases. It provides the hardware initialization, hardware control, host communications, and the DSP library functions that you need to develop VXI system applications. Examples are also provided to help you in your development process.

The VX8 Support Software includes two libraries: the Host Library (VX8 Instrument Driver) and the C4x DSP Library (VX8 C4x Support Software Library.)

Host Library The Host Library provides an easy-to-use host interface to the VX8 board and its TIM-40 modules. The software allows you to manage individual boards as well as VX8 systems. It provides the system initialization, code download, control functionality, and data transfer between the host and VX8 Carrier Board.

C4x DSP Library The VX8 C4x Support Software Library provides C4x-callable functions that optimize data transfers and simplify the task of configuring and controlling the VX8. Supported functionality includes:

- VXIbus routines (sCV64 DMA configuration and direct VXI master transfers),
- Global Bus routines (accesses from a DSP to the Global Shared or DRAM Shared Bus)
- HP Local Bus routines (bus configuration and data transfer),
- Node B DUART initialization, data transfer, and interrupt routines,
- user defined LED control,
- and Node A initialization Kernel routine (VXIbus A16 register access from the host and VX8 initialization after a board reset.)

PEROM programming utilities are also available to verify board level functionality at power up.

2 Hardware Installation and Configuration

This chapter describes the following hardware installation and configuration procedures:

- TIM-40 module installation
- DRAM memory SIMM installation
- Jumper configuration
- A16 Logical Address configuration (DIP switch S1)
- VX8 Carrier Board installation into chassis (before you install the VX8 board into a VXI chassis, you'll want to configure or install the above items.)

Most of the configuration procedures mentioned above require the top shield covering the top of the VX8 printed circuit board (PCB) to be removed. You can, however, configure the DIP switch (S1) without removing the top shield (see the section *Setting the VXIbus A16 Logical Address*.)

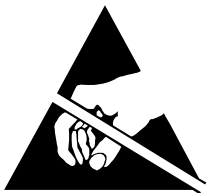
To install TIM-40 modules, DRAM, or configure the VX8 jumpers remove the top shield according to the steps in the procedure *Removing the Top Shield*.

To set the A16 Logical Address on DIP switch S1 go directly to the procedure *Setting the VXIbus A16 Logical Address*.

If you're adding DRAM memory or TIM-40 modules to your VX8 after you've installed the VX8 into your VXI mainframe, you'll have to first remove the VX8 from the mainframe. See the procedure *Removing the VX8 from a VXI Mainframe*.

Before you start You'll need the following tools:

- Phillips #1 screwdriver
- Grounding wrist strap
- Flat-tipped screwdriver
- Nut driver (for metric M3 size nuts)



Your computer, VX8 Carrier Board, and other modules can be damaged by static electricity. When handling modules:

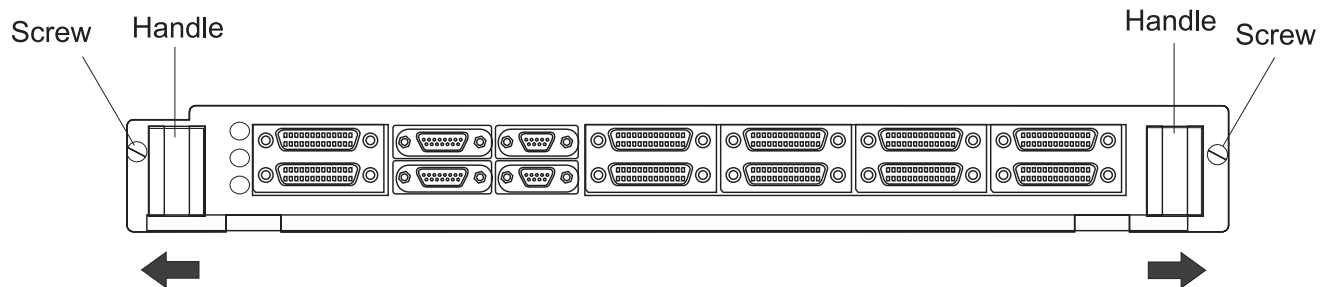
- Use a grounding wrist strap and follow the instructions provided with the strap.
- Do not wear clothing that is subject to static charge buildup, such as wool or synthetic material. Cotton is best for this type of work.
- Do not handle integrated circuits in carpeted areas.
- Do not remove any device from its anti-static bag until you're ready to install it.

- Avoid touching circuit leads as much as possible. In particular, avoid touching the front panel connectors of the VX8 Carrier Board.

2.1. Removing the VX8 from a VXI Mainframe

If you're adding DRAM memory or TIM-40 modules after you've installed the VX8 into your VXI mainframe, you'll have to first remove the VX8 from the mainframe. To remove a VX8 from a VXI Mainframe:

1. Shut down the operating system as described in the documentation provided with your controller.
2. Turn off the power for the VXIbus mainframe.
3. With a flat-tipped screwdriver, remove the two screws at the corners of the VX8 front panel.
4. Apply outward pressure to the handles on the front panel to disconnect the VX8 from the backplane connectors and holding the handles pull the board out of the mainframe. Place the board on a firm, static-free surface.

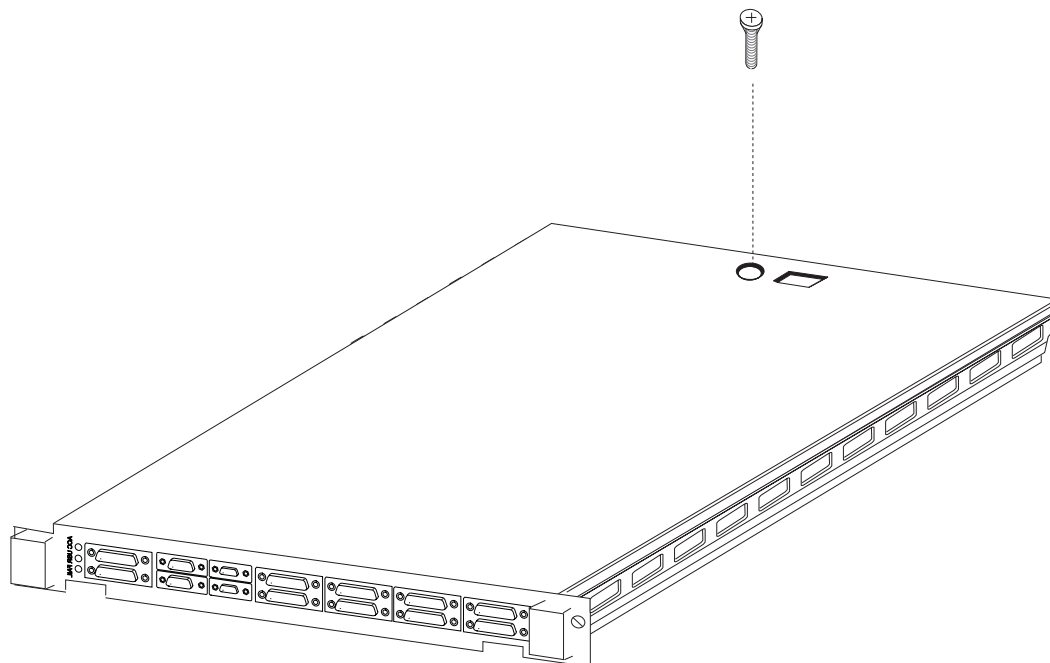


2.2. Removing the Top Shield

Caution: Do not remove the bottom shield of the VX8. There are no user configurable components on the bottom of the board.

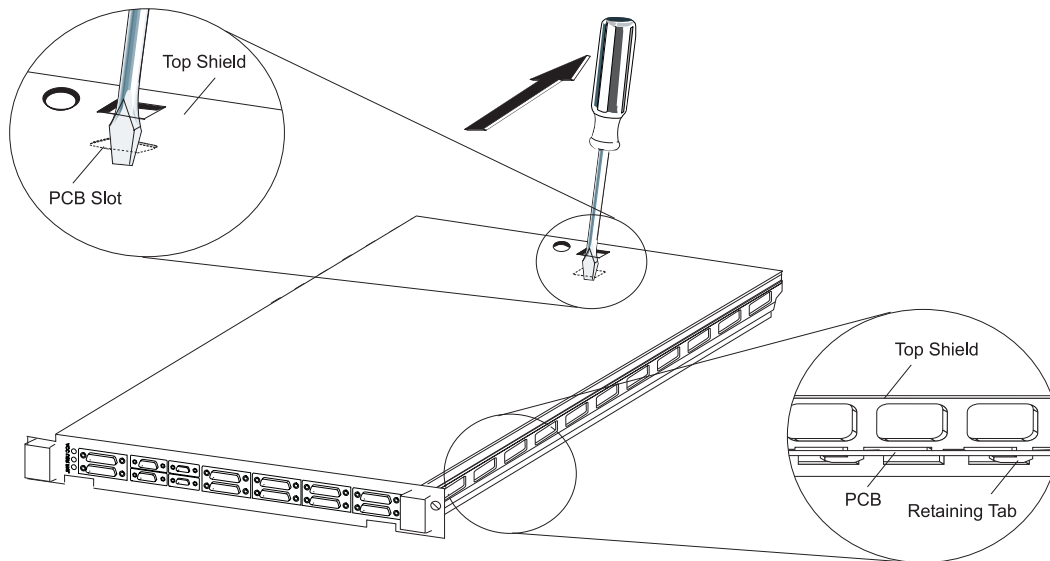
To remove the top shield from the VX8, follow these steps:

1. Remove the screw located at the back of the VX8 using a Philips #1 screwdriver.

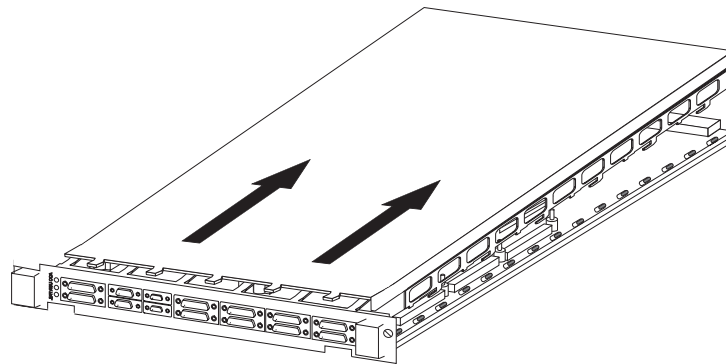


2. Insert a flat-bladed screwdriver in the slot of the top shield at the back of the unit as shown in the next figure. Ensure that the tip of the screwdriver's blade is inserted into the corresponding slot in the PCB.

3. Using the screwdriver as a lever, ease the top shield back so that the shield's retaining tabs clear their corresponding slots on the PCB. Do not use excessive force.



4. Tipping the back of the top shield away from the PCB, lift the top shield from the VX8.



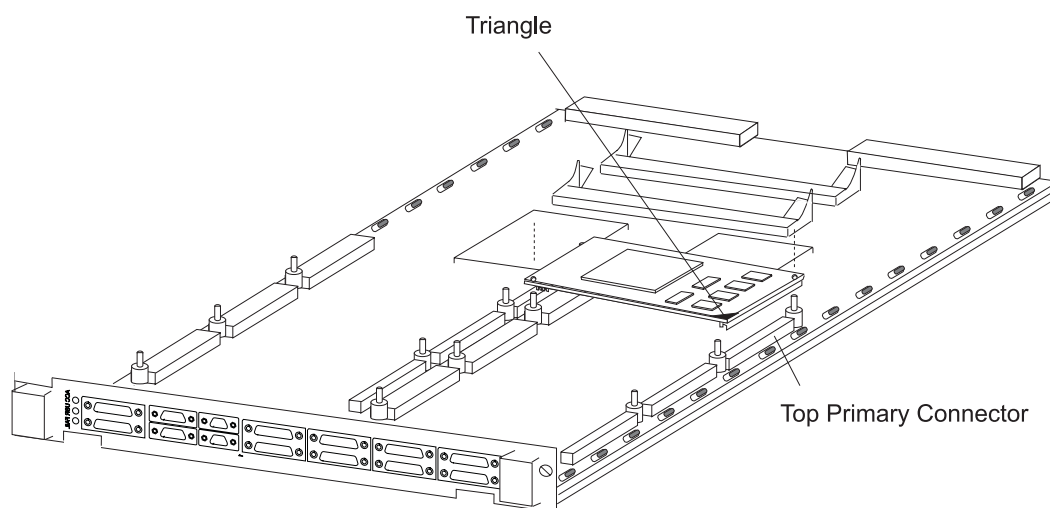
You can now install TIM-40 modules or VX8 jumpers according the procedures described in the following sections.

2.3. Installing TIM-40 Modules

The VX8 has six TIM-40 sites. To install a TIM-40 module onto a site follow these steps:

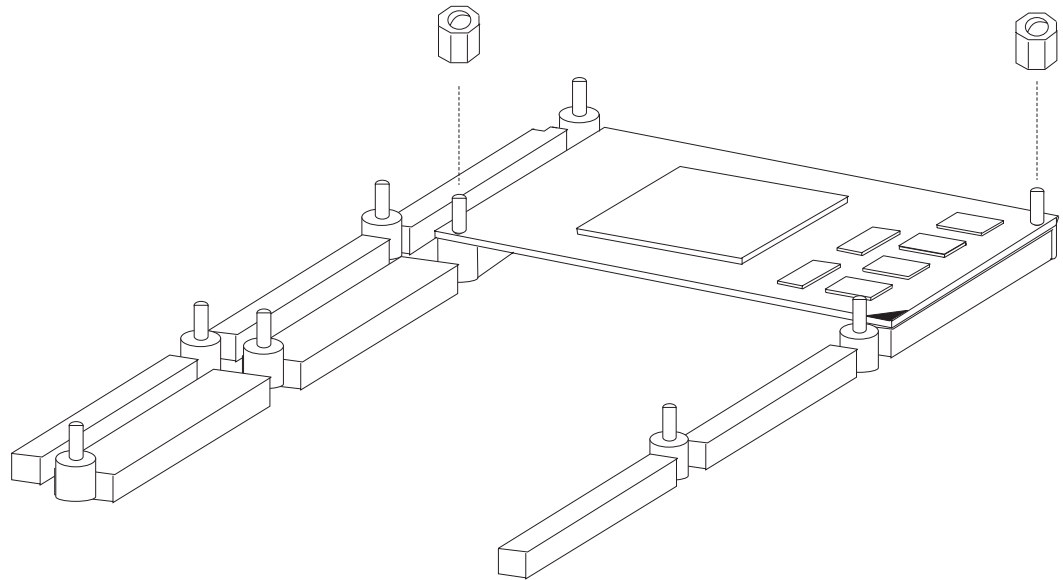
Note: If a TIM-40 module requires Global Bus access, ensure that the module is configured to use the external clock on the VX8 board and not the oscillator on the TIM-40 module. Refer to the particular TIM-40 module User's Guide for information regarding external clock configuration.

1. If you've already installed the VX8 into your VXI mainframe, follow the procedures described in *Removing the VX8 from a VXI Mainframe*.
2. Remove the top shield from the VX8 as described in the section *Removing the Top Shield* and locate the TIM-40 site on the VX8 in which the module will be installed.
3. Align the module over the TIM-40 site so that the triangle printed on the top right corner of the module PCB is adjacent to the top primary connector of the carrier board.



4. Ensure that the connectors and nylon retaining posts are properly aligned, and gently push the module on to the connectors. When pushing the module onto the connectors, apply pressure directly over the connectors to avoid bending the module.

5. Using a nut driver, fasten the module to the VX8 with the nylon nuts provided. There are two holes for these nylon screws at diagonal corners of the TIM-40 site. Do **not** over-tighten.



2.4. DRAM Memory Requirements

The VX8 Carrier Board supports up to 2 Banks of shared DRAM in its two 72-pin SIMM sockets, SIMM 1 and SIMM 2. The SIMM sockets, which can be left unpopulated, support both 4 and 32 Mbyte 60 ns SIMMs. The following types of DRAM memory modules can be used:

- 60ns Fast Page Mode 1Mx32 or 1Mx36 PC DRAM SIMMs
- 60ns Fast Page Mode 8Mx32 or 8Mx36 PC DRAM SIMMs

It is important to use DRAM SIMMs that meet the timing requirements of the VX8 Carrier Board. Use one of the following Micron DRAM SIMMs or an equivalent:

Manufacturer	4 Mbyte	32 Mbyte
Micron	MT8D132M-6	MT16D832M-6

4 Mbyte and 32 Mbyte SIMMs cannot be mixed on the same VX8. For example, you cannot put a 4 Mbyte SIMM in SIMM 1 and a 32 Mbyte SIMM in SIMM 2.

2.5. Installing DRAM SIMMs

As previously described, two sizes of SIMM memory can be used: 1M x 32 (4 Mbyte) or 8M x 32 (32 Mbyte). Three jumpers (JP3, JP4, and JP5) must be set according to the size of memory used in the sockets. The configurations that the DRAM memory can be installed in the two SIMM sockets and the jumper settings for each configuration are shown in the following table.

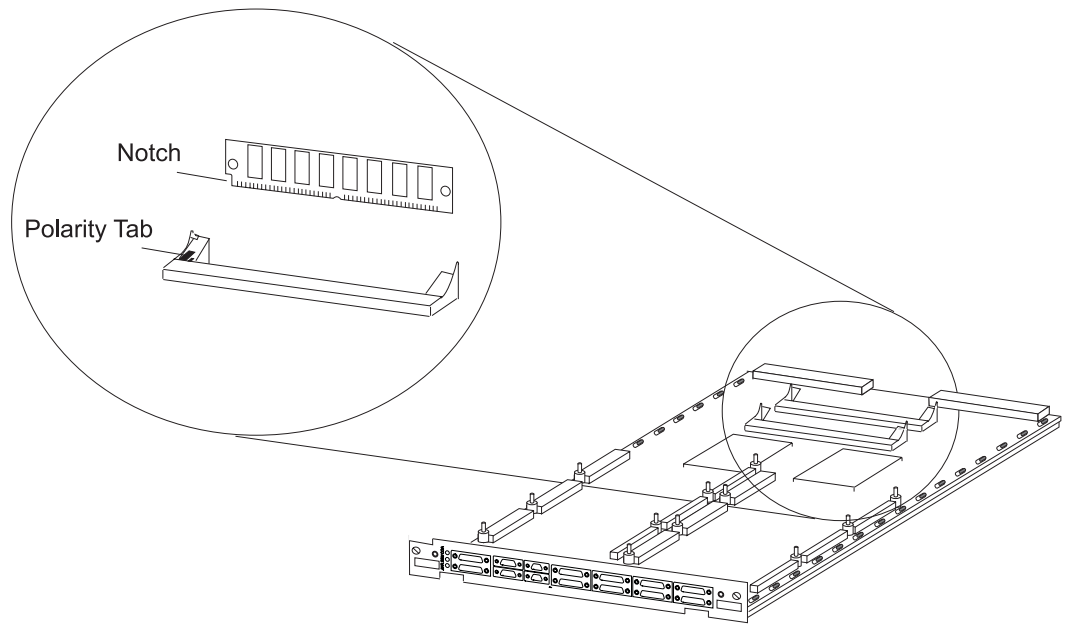
Table 1 DRAM Sizes and Jumper Settings

SIMM 1	SIMM 2	Total DRAM	JP3	JP4	JP5
-	-	0 Mbyte	1M	1M	1M
4 Mbyte	-	4 Mbyte	1M	1M	1M
4 Mbyte	4 Mbyte	8 Mbyte	1M	1M	1M
32 Mbyte	-	32 Mbyte	8M	8M	8M
32 Mbyte	32 Mbyte	64 Mbyte	8M	8M	8M

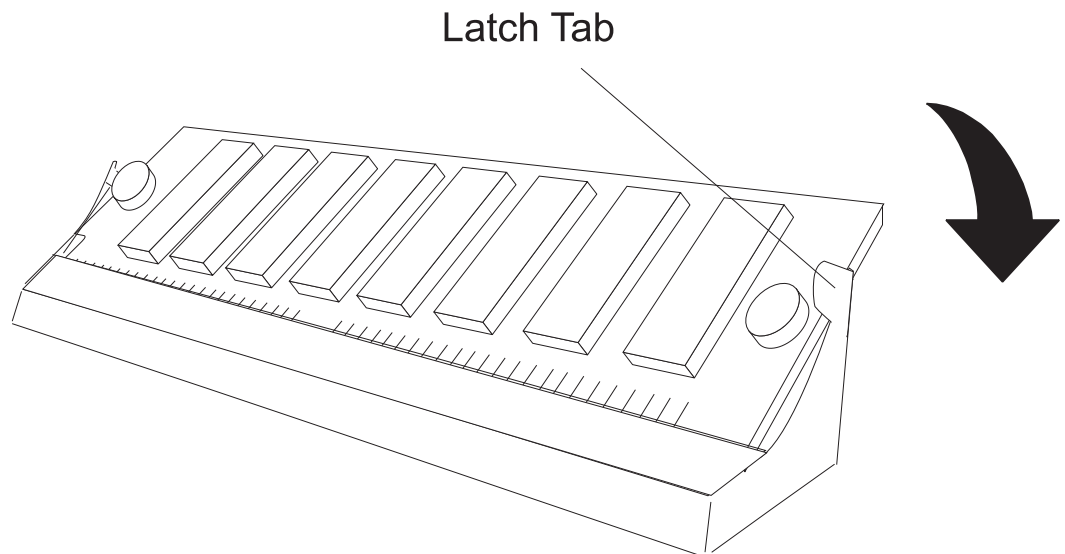
Note: If only one SIMM module is being installed, it should be installed in the SIMM 1 socket.

To install DRAM memory modules into the VX8 follow these steps:

1. If you've already installed the VX8 into your VXI mainframe, follow the procedures described in *Removing the VX8 from a VXI Mainframe*.
2. Remove the top shield from the VX8 as described in the procedure *Removing the Top Shield* and locate the DRAM memory SIMM sockets.
3. Grasp the DRAM module along its top edge and position it over the socket so that the notch on the SIMM module is aligned with the polarity tab in the lower left of the socket.



4. Insert the SIMM module into the socket at a 30° angle from vertical. Gently push it so that it is firmly seated into the socket.
5. Carefully rotate the SIMM module back and down until its edges snap behind the latch tabs of the socket. Apply equal pressure along the top of the SIMM module so that both edges engage the tabs at the same time.



6. Set jumpers JP3, JP4, and JP5 according to the type of memory installed as shown in the preceding table. Further information on jumper locations and settings is given in the next section.

2.6. Jumpers and DIP Switches

The following figure shows the location of the DIP switch and jumpers on the VX8.

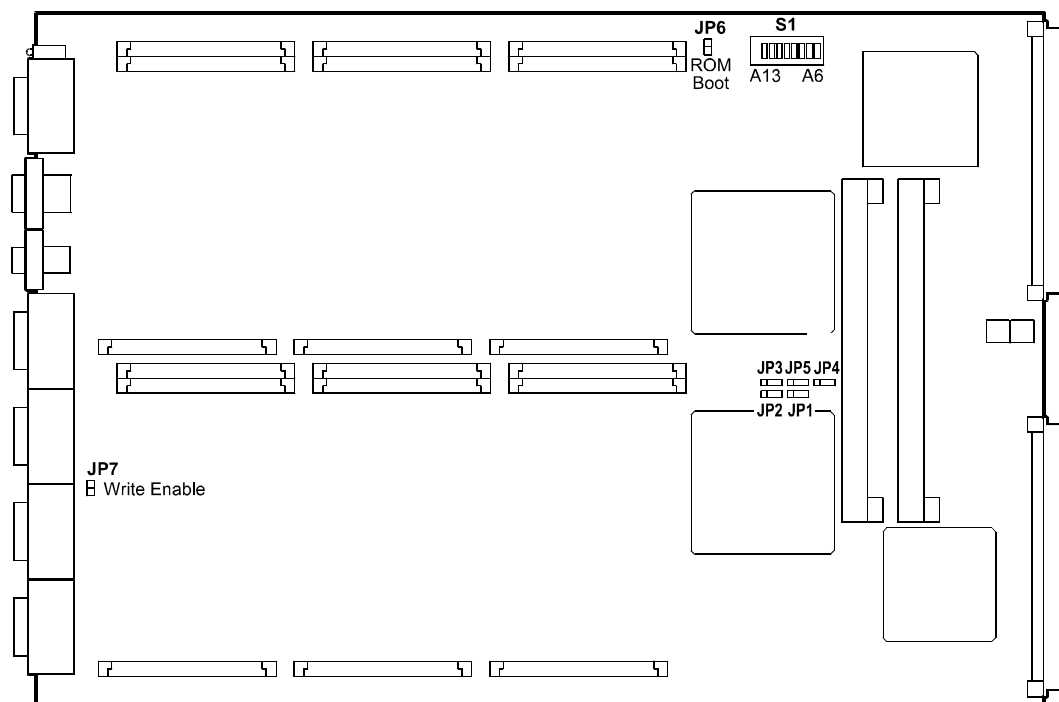


Figure 3 VX8 Jumper and Switch Locations

A summary of the Jumper settings is given in the following table.

Table 2 Boot Source Jumpers (JP1 and JP4)

Jumper	Pins	Function
JP1	1-2	TCLK1 is TTL buffered CLK10
	2-3*	TCLK1 is ECLKTRG1 (SYNCLK)
JP2	1-2*	ECLKTRG0 and ECLKTRG1/CLK10 buffers disabled
	2-3	ECLKTRG0 and ECLKTRG1/CLK10 buffers enabled
JP3 to 5	1M	Set all for 1 M x 32-bit DRAM SIMMs
	8M*	Set all for 8 M x 32-bit DRAM SIMMs
JP6	OUT*	Node B C40 boots from first active communication port
	IN	Node B C40 boots from local bus PEROM
JP7	IN	Node A Boot Kernel PEROM Write Protection disabled
	OUT*	Node A Boot Kernel PEROM Write Protection enabled

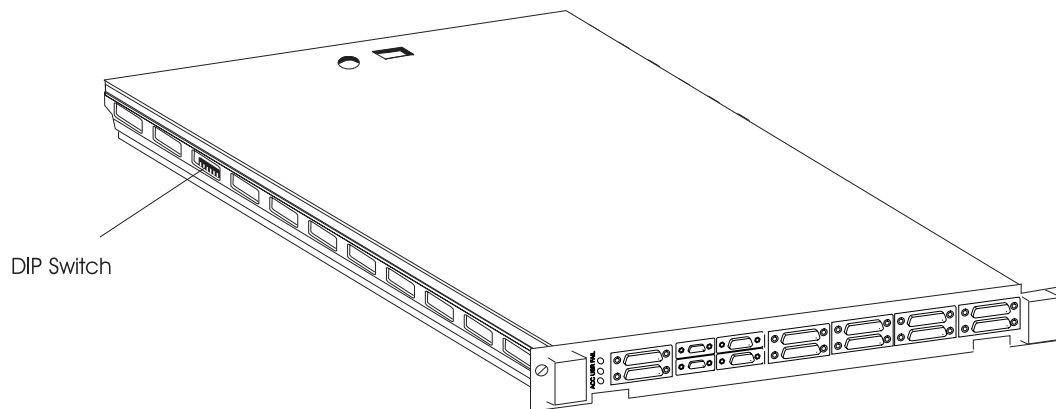
* Default setting

2.6.1. Setting the VXIbus A16 Logical Address

The VXIbus A16 Logical Address is set by DIP Switch S1. You can access this DIP switch through the openings in the side of the top shield—without removing the top shield cover.

To set the A16 address:

1. If you've already installed the VX8 into you're VXI mainframe, follow the procedures described in *Removing the VX8 from a VXI Mainframe*.
2. Set the switches on the DIP switch to the desired A16 address according to the following figure and table. Use a small screwdriver, or similar tool, to reach through the hole in the side of the shield.



The default address is read as (MSB..LSB) in binary, so the value shown below is read as 11110000 binary or 240 decimal. It represents the 240th logical address in the A16 space. Note that the switch is “0” when ON, and “1” when OFF.

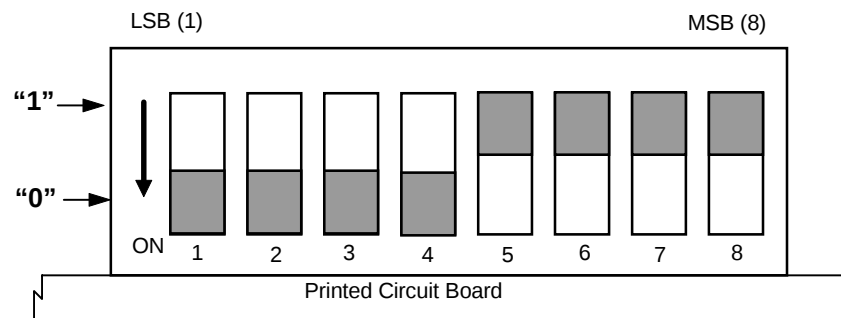


Figure 4 A16 Address DIP Switch

Note: Ensure that no two devices are configured to use the same A16 logical address.

Table 3 DIP Switch A16 Addresses

Pos.	DIP Switch Positions 4321															
8765	0000 0001 0010 0011				0100 0101 0110 0111				1000 1001 1010 1011				1100 1101 1110 1111			
0000	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0001	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0010	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
0011	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
0100	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
0101	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
0110	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
0111	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
1000	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
1001	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
1010	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
1011	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
1100	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
1101	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
1110	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
1111	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

* Default setting

Observe these precautions for the following A16 address:

Address	Precaution
0	Do not use. This address is reserved for the VXIbus Resource Manager (RM).
1 to 15	Should not be used. These addresses are typically reserved for mainframe extension devices.
255	Do not use this address in dynamically configured systems.

2.6.2. Selecting the Node B Boot Mode

The Node B TMS320C40 DSP can be set to boot from either the first active communication port or from its Local Bus PEROM, by jumper JP6.

Jumper JP6 is used to select whether Node B boots from Communication Port (JP6 removed) or from Local Bus PEROM (JP6 installed). The default is for booting from Communication Ports. Node B is integral to booting and loading code to all processors on a VX8 Carrier Board. This jumper should only be configured for booting from PEROM if an embedded solution (no download from host) is required, or if the booting method does not use the Spectrum supplied drivers.

2.6.3. Setting the Node A Kernel PEROM Write Enable

Jumper JP7 is used to prevent accidental writes to Node A's Boot Kernel PEROM. The default is for write protect, and the jumper is removed. If the jumper is installed, then the PEROM can be erased and overwritten.

Note: A second PEROM is provided for TIM-40 compatibility (IDROM) or for use by the application.

Caution: Do **not** alter the contents of the Boot Kernel PEROM unless you are re-programming it with a firmware update supplied by Spectrum. The code contained in this PEROM is mandatory for proper functioning of the VX8 Carrier Board. If this PEROM is accidentally erased or corrupted the VXI resource manager will assume the board is dead and will place the VXI in reset via the reset bit in the A16 register. You will then have to manually take the board out of reset and use an XDS to reprogram the Boot Kernel PEROM through the front panel JTAG.

2.6.4. DRAM Memory Installation

Jumpers JP3, JP4, and JP5 select the size of the DRAM SIMMs that are installed in the VX8 Carrier Board. If no DRAM SIMMs are used, then the position is irrelevant.

Note: All three jumpers must be changed at once, and all three jumpers must be in the 1M position if 1Mx32 (4 MBytes) SIMMs are used or in the 8M position if 8Mx32 (32 MBytes) SIMMs are used.

Refer to the *DRAM Shared Bus* section of the *VX8 Carrier Board Technical Reference Manual* for further details regarding DRAM SIMMs. The base configuration is for no SIMMs installed, and the jumpers are installed in the 8M position.

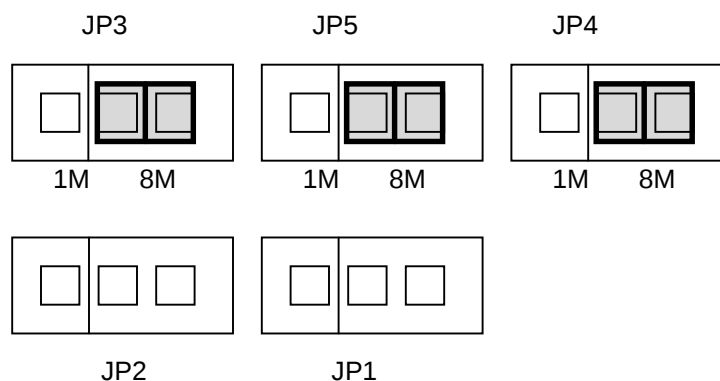


Figure 5 DRAM Jumper Settings

2.6.5. TIM-40 TCLK1/UDP12 Sources

Jumpers JP1 and JP2 control the functionality of the VXI defined ECL signals CLK10, ECLTRG0 and ECLTRG1. These ECL signals are TTL buffered on the VX8 Carrier Board, and are routed to the TIM-40 module TCLK1 (timer pin 1) and User Defined Pin 12. The TTL buffered version of ECLKTRG0 (SYNDAT) is brought to a clock buffer.

The source of TCLK1 is selected by JP1.

- To route the TTL buffered CLK10 to TCLK1 through a clock buffer, install a jumper in position 1-2.
- To route ECLTRG1 (SYNCLK) to TCLK1 through a clock buffer, install a jumper in position 2-3. This is the default setting.

The buffers that provide CLK10, ECLTRG0 and ECLKTRG1 are enabled or tri-stated depending upon JP2.

- To disable the ECLKTRG0 and ECLKTRG1/CLK10 buffers, install a jumper in position 1-2. This is the default setting.
- To enable the ECLKTRG0 and ECLKTRG1/CLK10 buffers so that they are driven to the TIM-40 modules, install a jumper in position 2-3.

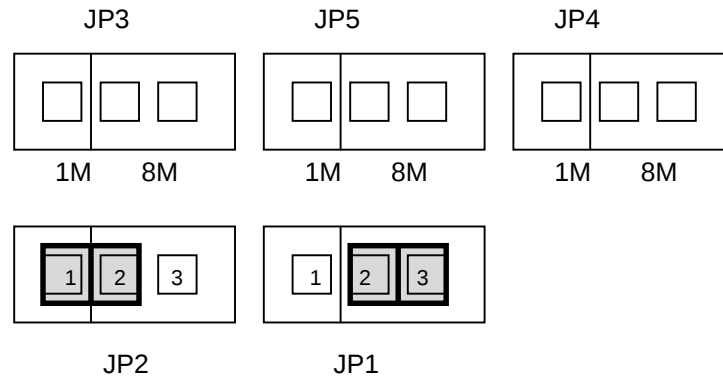
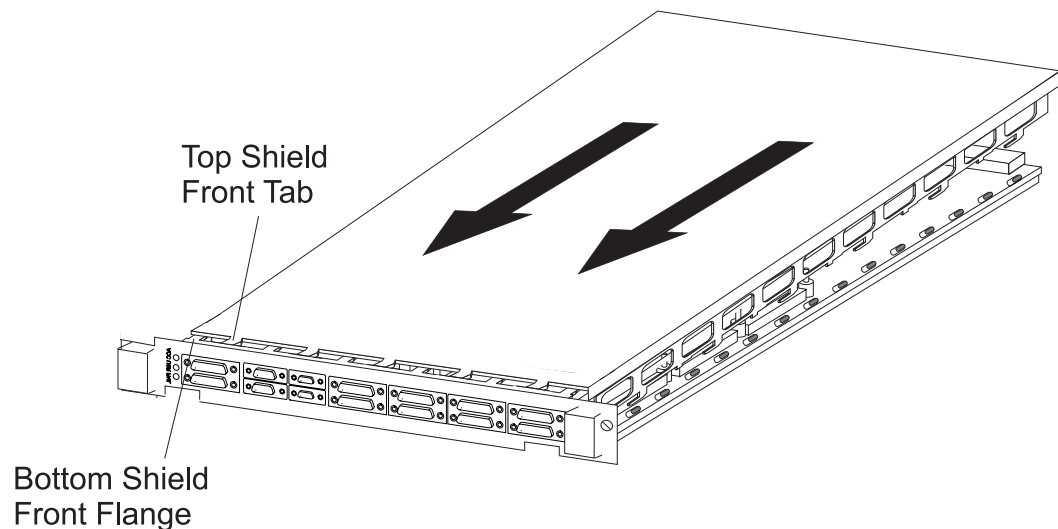


Figure 6 CLK10, ECLTRG0 and ECLTRG1 Jumpers

2.7. Replacing the Top Shield

To replace the top shield on to the VX8, follow these steps:

1. Position the shield so that the front tabs slide under the front flange of the bottom shield attached to the VX8. Ensure the tabs of both shields mesh.



2. Slide the shield forward and tip it downward so that that the tabs on the side of the shield are inserted into the slots in the PCB.
3. Using the screwdriver as a lever, ease the top shield forward so that the shield's retaining tabs are seated in their corresponding slots on the PCB. Do not use excessive force.
4. Replace the screw in the back of the cover.

You can now install the VX8 into a VXI rack according to the following procedure.

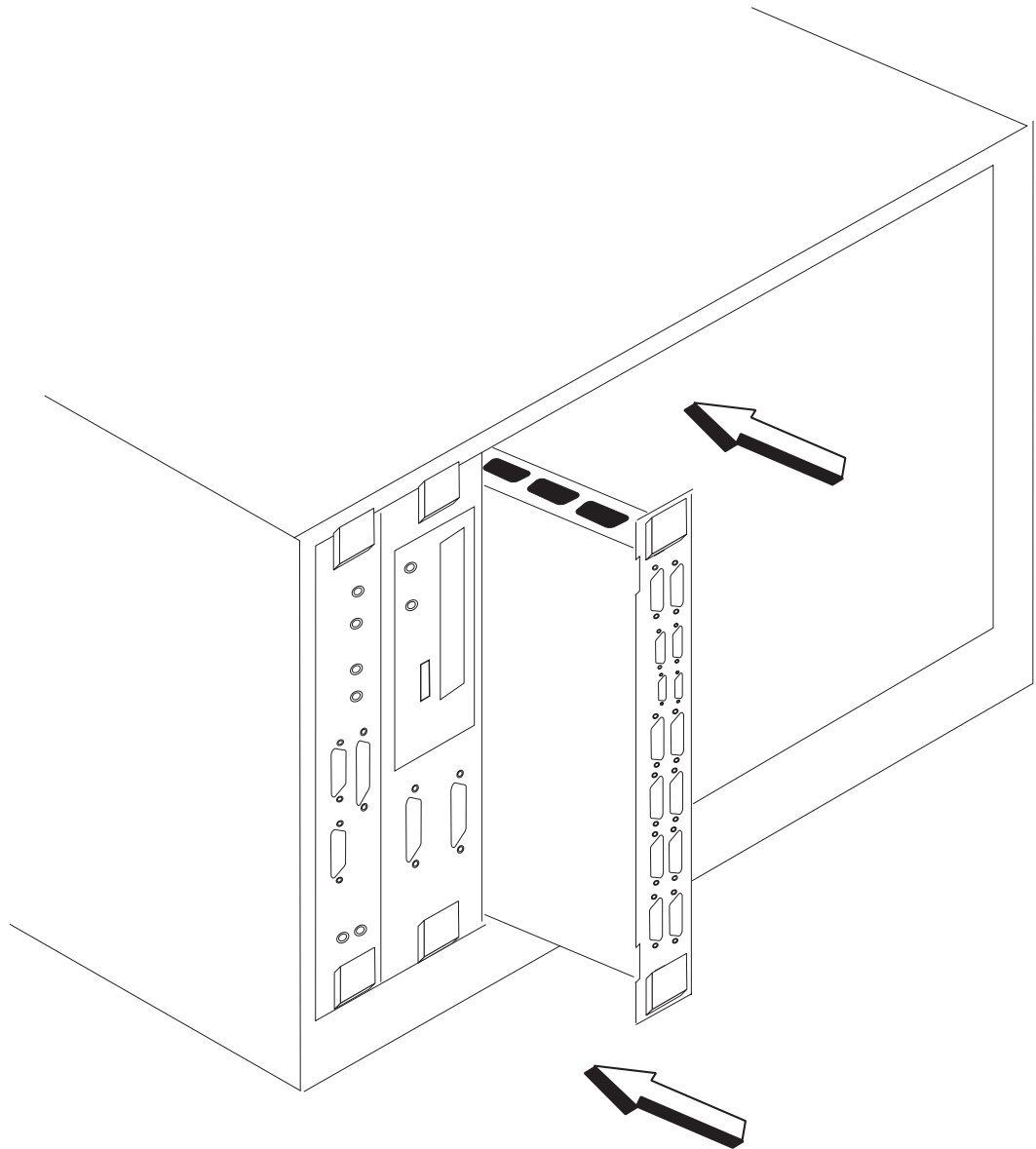
2.8. Installing the VX8 in a VXI Mainframe

Caution: The HP Local Bus specification for the VX8 Carrier Board uses ECL signaling levels on both the A and C sides. Only ECL compatible modules can be installed adjacent to the VX8 Carrier Board or the system could be damaged.

The VX8 module is keyed for this VXI implementation, do not force modules that are not keyed for this implementation into adjacent slots.

Only install the VX8 Carrier Board in a VXI rack with the power OFF.

1. Power down the VXI mainframe.
2. Select a slot in the mainframe for the VX8 Carrier Board. Note that the HP Local Bus Interface is only routed to adjacent slots. A data source sending data to the VX8 must be installed on the A side (left). Data sinks receiving data generated by the VX8 must be installed on the C side (right).
3. Configure the VX8 board. Ensure that TIM-40 modules, memory modules and user defined jumper settings (especially VXIbus A16 Logical Address) are configured and properly installed.
4. Place the card edges of the VX8 Carrier Board into the module guides of the mainframe and slide the board into the chassis until the board connects firmly with the backplane connectors. If your board is equipped with long handled ejectors, then make sure that the ejectors are properly seated on the mainframe before final board insertion.



5. Install the mounting screws to attach the front panel of the VX8 Carrier Board to the mainframe.

3 VX8 C4x Support Software Installation

The disk included with the VX8 Developer's Kit contains support functions and example programs for programming the VX8 Carrier Board.

The included files are used with the TI (Texas Instruments) assembly and "C" language tools, and also the TI debugger, to develop applications for the VX8 DSPs.

3.1. Directory Structure

The **setup.exe** program used to install the VX8 C4x Support Software sets up the following directory structure under the directory you specify.

Table 4 VX8 Developer's Kit Disk Contents

Directory/File	Description
\build	This directory provides the files required to recompile the VX8 C4x Support Software.
\examples	This directory provides "C" language source code examples that demonstrate how to perform interrupt handling and data transfers on the VXIbus, HP Local Bus, and the RS-232 DUART. readme.txt files are provided with each example. This directory also provides files that you can use to build your own software, including batch, linker, and C initialization files.
\include	This directory provides the Include files for the VX8. vx8.h is the main include file that should be included in your code.
\lib	This directory contains the C4x library and archive.
readme.txt	This file describes the contents of the VX8 Developer's Kit Disk.
\source	This directory provides the "C" language source code for the VX8 C4x Support Software functions.
A0002	This file name indicates the version number
\kernel	This directory contains the source for the comm LOAD and Node A kernels. These are supplied for reference and are not required by the user for normal use of the C4x library.
\examples.c44	This directory contains examples using the vx8c44ss.lib for use with TMS320C44 DSP-based TIM-40 modules. The example code is the same as in \examples , except that vx8c44xx.lib is linked instead of vx8c40ss.lib .

3.2. Requirements

You should be familiar with VISA and/or SICL software development. A knowledge of ANSI C software development on VME/VXIbus systems, and TMS320C4x software development in ANSI C/assembly is vital.

VX8 Support Software for HP-UX 9.X/SICL was developed on an HP V743 Embedded controller with HP-UX 9.05 and SICL C.03.09. This version of the VX8 Support Software is to be used with the following hardware and software configurations:

Hardware	Software
• VXI mainframe chassis with a minimum of 2 slots (for slot 0 controller and a single VX8 card) • VXI slot 0 controller (HP V743 VXI Embedded Controller for example) • VX8 Carrier Board • DOS/WIN 95 PC and an external JTAG interface for TMS320C4x DSP software development	• ANSI-C compiler • DOS based TI TMS320C4x development tools • Windows 95 or Windows NT (Windows NT requires TI Tools version 5.0)

The VX8 VISA Windows 95 and Windows NT Instrument driver supports the following hardware and software configurations:

Hardware	Software
• VXIbus mainframe chassis with a minimum of 2 slots (for slot 0 controller and a single VX8 card) • VX8 Carrier Board • DOS or Windows 95 PC with an external JTAG interface for TMS320C4x DSP software development. This can be the same PC as the host if running Windows 95. TI Tools version 5.0 and Go DSP Code Composer are required for 'C4x development on Windows NT.	• Microsoft Visual C compiler version 5.0 • TI TMS320C4x development tools (version 5.0 required for Windows NT) • VISA (version 1.1 or later) • Windows 95 or Windows NT

3.3. Installing the VX8 C4x Support Software

To install the VX8 C4x Support Software:

1. Insert the VX8 C4x Developer's Kit disk into a 31/2" floppy disk drive.
2. Run the **setup.exe** program from the VX8 C4x Developer's Kit disk.

3. Enter the path where you want the VX8 C4x Support Software. The default directory is C:\vx8c4xss.
4. Make sure that the paths for the VX8 C4x Support Software include files (C:\vx8c4xss\include) and library (C:\vx8c4xss\lib) are included in the C_DIR environment variable used by the TI C compiler. In addition, ensure that the path for the VX8 Support Software include files are also included in the A_DIR environment variable used by the TI C assembler. If you're using TI's debug monitor, make sure that the path for the C source files (C:\vx8c4xss\src) are included in the D_SRC environment variable if you wish to view the vx8c4xss code while debugging.

Note: If you have not already done so, you should install Toolbox at this time. Refer to the *Toolbox Configuration Utilities User Guide* for installation procedures.

3.4. Software Configuration Notes

Each VX8 must be set up with a non-contending VXIbus Logical Address (LA). It is assumed that there will be enough VXI/VME resources for each VX8 in the system.

You must configure the Resource Manager (RM) so that it correctly reflects the resources which the application software expects. This is particularly important with respect to the resources assumed by the SCV64 device on each VX8 in a system. The RM configuration files will provide a facility to force the assignment of interrupt handler levels (IEEE 1155 RULE 3.4.11 [3]).

Note: Since the VX8 does not have message based interrupt configuration capabilities, the interrupt levels and interrupt handler levels will have to be specified in the RM configuration files **as well as** in the VX8 configuration integrated within the DSP application program. The DSP application program must set up the interrupts on the SCV64 by calling the **VX8_SCV64EnableInterrupt** function in the C4x Support Software.

VX8 system configuration is managed through a hardware system definition file (SDF) and a software load definition file (LDF). It will be left up to you to ensure that the VX8 hardware configuration matches the configuration expected by your software application.

The DSP application software will be required to do several things:

- All DSP code should de-assert the TIM-40 defined /CONFIG line after boot and wait for all nodes to be loaded before proceeding with normal operations. De-assertion of /CONFIG is normally taken care of by the C initialization code (boot_a.asm, boot_b.asm, and boot_tim.asm.) IIOF3 is generally used for /CONFIG on Spectrum's TIM-40 modules.
- If VXIbus interrupts are required, the DSP application software must set up the interrupt handler levels and interrupter level on the SCV64 and these levels must

correspond to the levels defined in the RM configuration files for the VX8 in question. Initial SCV64 set up and A16 Register servicing will be carried out by the Node A Embedded C40. SCV64 interrupt capability should not be enabled until the end of the VX8 load process since DSP applications will not be ready to handle interrupts until software loading has completed.

- Ensure that COMM port transfers are not initiated until the loading process is complete. This will prevent DSPs that are attempting to boot via COMM port from being corrupted. We suggest the use of a flag by the Host to signal the DSP to resume code execution.

When building the DSP application software for the Node A DSP, the A16 ISR (BOOT_IIOF3Isr) for the VX8 must be linked in and mapped as the IIOF3 ISR in the interrupt vector pointer. The VX8 will not respond to VXI A16 configuration register commands without the Node A A16 ISR resident. This is performed automatically by the **boot_a.asm** C initialization code.

Note: When building code, you must link the supplied **boot_?.obj** first. When building code under TI Tools version 5.0, use the -v0 flag to force COFF version 0 output files. COFF version 2 will be supported in a future release.

3.5. Using the Example Code

The example programs are included on VX8 C4x Support Software development disk. Refer to the **readme.txt** files provided with each example for instructions on setting up and running the examples.

To verify that you have installed the VX8 Carrier Board and Support software, run the **mult.c** example found in the **mult** directory of the VX8 C4x Support Software development disk.

If you are using 'C4x based TIM-40 modules, you should run the examples in the **examples.c44** directory. These have been linked to **vx8c44ss.lib**.

The examples are also documented in the *VX8 Carrier Board Programming Guide*.

4 VX8 Instrument Driver Installation

The instrument driver provides host applications with a software interface to basic routines for configuring, controlling, and communicating with the VX8. There are two supplied versions of the instrument driver, one for SICL I/O libraries and one for VISA I/O libraries. Both versions are written in standard ANSI C, both versions support multiple VX8s in a system, and both versions allow the host to open multiple sessions to any single VX8.

4.1. HP-UX 9.0x SICL Instrument Driver Installation

The following section details the installation procedure for the HP-UX 9.05 SICL instrument driver.

4.1.1. Tape Contents

The contents of the VX8 Support Software tape are contains the following fileset:

Table 5 VX8 Developer's Kit Tape Fileset

File	Description
SSVX8-A_00-00	SSVX8 Programming environment, A.00.01

When installed, the VX8 Support Software tape will generate the directory structure described in the following table. Please refer to the **readme.txt** files found in the `/opt/vxipnp/hpux/ssvx8/host` and `/opt/vxipnp/hpux/ssvx8/dsp` directories for more information about the directory contents.

Table 6 VX8 Developer's Kit Tape Installed Directory Structure

Directory Structure	Description
/opt/vxipnp/hpux	- HP-UX VXIpnnp root directory
/bin libssvx8.sl	- Bin directory containing shared libraries - VX8 Instrument Driver shared library
/include ssvx8sys.h ssvx8.h	- Top level include files from installed VXI software components - VX8 Instrument Driver header file - VX8 Instrument Driver header file for VXIpnnp components
/lib libsdf.a libssvx8.a	- Archive (static) libraries from installed VXI software components - Spectrum Signal Processing System Definition File library - VX8 Instrument Driver library
/ssvx8	- VX8 source root directory
/host readme.txt libsdf.a libssvx8.a libssvx8.sl makefile /examples /mult /include /lib /lst /src	- VX8 Instrument Driver source root directory - readme file about the VX8 Instrument Driver directory contents - Spectrum Signal Processing System Definition File library - VX8 Instrument Driver library - VX8 Instrument Driver shared library - VX8 Instrument Driver makefile - Host examples - Multiplication example - VX8 Instrument Driver Include directory - VX8 Instrument Driver Object directory - VX8 Instrument Driver Listing directory - VX8 Instrument Driver Source directory
/dsp /build /examples /examples.c44 /include /kernel /lib /src	- VX8C4x Support Software root directory - VX8C4x Support Software build directory - 'C40 DSP examples directory - 'C44 DSP examples directory - VX8C4x Support Software Include directory - VX8C4x Support Software kernel software directory - VX8C4x Support Software Library directory - VX8C4x Support Software Source directory

4.1.2. Requirements

You must have SICL (Standard Instrument Control Library) installed on your system before the VX8 SICL Instrument Driver can function.

4.1.3. Installing the VX8 SICL Instrument Driver from Tape

The host system for loading the VX8 Host Support Software tape must be a HP-UX 9.0x workstation.

You will also need a Digital Data Storage (DDS) tape drive to read the contents of the DDS tape.

In order to read the tape, you must have **update** (an interactive program used to load software from local or remote media) installed on your system.

To install the software:

1. Become super user (root).
2. Write-protect the tape by sliding the write-protect tab on the cartridge to the appropriate position.
3. Insert the tape cartridge in your tape drive.
4. Enter the following command to start Update:

`/etc/update`
5. The default source media address is `/dev/update.src`. To specify a different device file, select the following menu item:

`Change Source or Destination ->`

then select the following:

`From Tape Device to Local System ...`

and follow the instructions that appear on your screen. Make sure that the default destination directory (/) is specified.
6. After the source and destination are set up, select:

`Select/View Partitions and Filesets ...`
7. Select the SSVX8 partition.
8. Once you have selected the SSVX8 partition, select:

`Start Loading`

The VX8 Host Support Software is loaded onto your system. The VX8 software is installed in the `/opt/vxipnp/hpux` directory. The instrument driver source and VX8 C4x

Support Software files can be found in the `/opt/vxipnp/hpux/ssvx8` directory. Please refer to the **readme.txt** files found in `/opt/vxipnp/hpux/ssvx8/host` and `/opt/vxipnp/hpux/ssvx8/dsp` for more information about the directory contents.

To remove filesets from your HP-UX workstation, invoke the `/etc/rmf` utility. Select the VX8 fileset with a “y” and then select “Begin Removal” to remove the VX8 Support Software from your system.

4.2. Windows 95/WinNT VISA Instrument Driver Installation

This section describes how to install the Windows95 / WinNT VISA instrument driver. The VX8 Instrument driver contains functions which will allow your host application software, configure, control, and communicate to a system of VX8 devices.

4.2.1. Installation Description

The VX8 Windows95/WinNT VISA Instrument Driver installation will install the software into several directories on your computer. If you have a VISA I/O library installed, the instrument driver header file, library, and DLL will be copied into the VXIPNPPATH directory. If VXIPNPPATH is not set up (i.e. VISA is not already installed), the instrument driver header file, library, and DLL will be copied into `c:\vxipnp\win95` or `c:\vxipnp\winnt` depending on your OS as permitted by the VXIpn Windows95 and WinNT frameworks.

Table 7 VX8 Instrument Driver VXIpn Installed Files

File	Description
[VXIPNPPATH]\kbase\ssvx8.kb	VX8 Knowledge Base File
[VXIPNPPATH]\winnt\bin\ssvx8_32.dll	VX8 Instrument Driver dynamic link library
[VXIPNPPATH]\winnt\include\ssvx8.h	VX8 Instrument Driver header file
[VXIPNPPATH]\winnt\lib\msc\ssvx8_32.lib	VX8 Instrument Driver dynamic link library

The VX8 Instrument Driver Source code and include files will be installed into the directory chosen during the installation. The default directory is [VXIPNPPATH]\win95\ssvx8 or [VXIPNPPATH]\winNT\ssvx8 depending on your OS.

Table 8 VX8 Instrument Driver Source Directory Structure

File	Description
bin\	DLLs

File	Description
build\	This directory provides the files required to recompile the VX8 Instrument Driver.
examples\	This directory provides “C” language source code examples.
include\	This directory provides the Include files for the VX8.
lib\	This directory contains the C4x library and archive.
license.txt	Spectrum source code license agreement
readmev.txt	This file describes the contents of the VX8 Instrument driver installation.
src\	This directory provides the “C” language source code for the VX8 C4x Support Software functions.
uninst.isu	Uninstall shield information

4.2.2. Requirements

You should be familiar with VISA and/or SICL software development. A knowledge of ANSI C software development on VME/VXIbus systems, and TMS320C4x software development in ANSI C/assembly is vital.

VX8 Instrument Driver for Win95 and WinNT frameworks was developed on an NI VXI-MXI2 environment and compiled under MSVC 5.0. No Borland support is currently available. This version of the VX8 Support Software is to be used with the following hardware and software configurations:

Hardware	Software
- VXI mainframe chassis with a minimum of 2 slots (for slot 0 controller and a single VX8 card) - VXI slot 0 controller (NI VXI-MXI2 for example) - VX8 Carrier Board - IBM compatible PC capable of running Windows95 or WIN NT - External JTAG interface for TMS320C4x DSP software development (might require a second PC if the previous PC does not have expansion slots).	- MSVC 5.0 - TI TMS320C4x development tools version 4.70 or 5.0 (5.0 required for WinNT DSP software development) - Windows 95 or WinNT 4.0 - VISA (VXIbnp VISA 1.1 or newer)

4.2.3. Installing the VX8 VISA Instrument Driver from Diskette

To install the VX8 Support Software:

1. Insert the VX8 Developer's Kit disk into a 3 1/2" floppy disk drive.
2. Run **setup.exe** on the VX8 Win95/WinNT VISA Instrument Driver disk. The install wizard will ask you to specify the VX8 Instrument Driver source directory. Under WinNT, you may require an account with administrator privileges. Installshield will then install the VX8 Instrument Driver files into the specified directory.
3. If you haven't already, install the VX8 C4x Support Software as described in a previous section.

4.2.4. Software Configuration Notes

The VX8 Instrument Driver was compiled as a Win32 DLL under MSVC 5.0. The compiler environment on the host should be set in the following manner:

ssvx8.h is the main include file that should be included in your application software.

Additional Include Directories (for default installation):

For WinNT:

[VXIPNPPATH]\WinNT\include

[VXIPNPPATH]\WinNT\ssVX8\include

For Windows 95:

[VXIPNPPATH]\Win95\include

[VXIPNPPATH]\Win95\ssVX8\include

Additional Preprocessor Definitions:

_VISA

Additional Library paths:

For WinNT:

[VXIPNPPATH]\WinNT\lib\msc

For Win95:

[VXIPNPPATH]\Win95\lib\msc

4.2.5. Hardware Configuration Notes

The VX8 must be set up with a non-contending VXIbus Logical Address (LA). It is assumed that there will be enough VXI/VME resources for each VX8 in the system.

You must configure the Resource Manager (RM) so that it correctly reflects the resources which the application software expects. This is particularly important with respect to the resources assumed by the SCV64 device on each VX8 in a system. The RM configuration files will provide a facility to force the assignment of interrupt handler levels (IEEE 1155 RULE 3.4.11 [3]).

Note: Since the VX8 does not have message based interrupt configuration capabilities, the interrupt levels and interrupt handler levels will have to be specified in the RM configuration files **as well as** in the VX8 configuration integrated within the DSP application program. The DSP application program must set up the interrupts on the SCV64 by calling the **VX8_SCV64EnableInterrupt** function in the C4x Support Software.

VX8 system configuration is managed through a hardware system definition file (SDF) and a software load definition file (LDF). It will be left up to you to ensure that the VX8 hardware configuration matches the configuration expected by your software application.

5 System Definition File Setup

The System Definition File (SDF) is a text based file which defines the hardware configuration of your VX8 System. The SDF modularly defines the hardware components in a logical manner. From top down, a VX8 System is composed of boards and COMM port connections. Boards are comprised of TIM-40 modules, COMM port connections, and a base address. TIM-40 modules are comprised of processors and their COMM port connections, and processors are comprised of attributes.

The SDFs provided by Spectrum should be modified using a text editor to specify the TIM-40 modules that are on your VX8 system. Future releases of the product will include a custom SDF editor.

For further details on the SDF and for examples of the different SDFs provided by Spectrum refer to the *System Definition File (SDF)* section of the *VX8 Carrier Board Programming Guide*.