

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*

SCRAMNet+ Network

SC150 VME6U
Hardware Reference

Document No. D-T-MR-VME6U###-A-0-A9

Systran

FOREWORD

The information in this document has been carefully checked and is believed to be accurate; however, no responsibility is assumed for inaccuracies. Systran reserves the right to make changes without notice.

Systran makes no warranty of any kind with regard to this printed material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

©Copyright 2003 Systran Corporation. All Rights Reserved.

SCRAMNet® is a registered trademark of Systran Corporation. U.S. Patent # 4,928,289.
VAX™ and VMS™ are trademarks of DIGITAL Equipment Corporation.
UNIX® is a registered trademark of UNIX System Laboratories, Inc., a wholly owned subsidiary of Novell, Inc.
ST® is a registered trademark of AT&T.

Revised: November 3, 2003

Systran Corporation
4126 Linden Avenue
Dayton, OH 45432-3068 USA
(800) 252-5601 (U.S. only)
(937) 252-5601

FCC

This product is intended for use in industrial, laboratory or military environments. This product uses and emits electromagnetic radiation, which may interfere with other radio and communication devices. The user may be in violation of FCC regulations if this device is used in other than the intended market environments.

CE

As a component part of another system, this product has no intrinsic function and is therefore not subject to the European Union CE EMC directive 89/336/EEC.

TABLE OF CONTENTS

1. INTRODUCTION 1-1
- 1.1 How To Use This Manual 1-1
  - 1.1.1 Purpose 1-1
  - 1.1.2 Scope 1-1
  - 1.1.3 Style Conventions 1-1
- 1.2 Related Information 1-1
- 1.3 Quality Assurance 1-2
- 1.4 Technical Support 1-2
- 1.5 Ordering Process 1-3

2. SCRAMNET NETWORK 2-1
- 2.1 Overview 2-1
- 2.2 Shared Memory 2-1
  - 2.2.1 Dual Port Memory Controller 2-1
  - 2.2.2 Control/Status Registers (CSRs) 2-1
  - 2.2.3 Virtual Paging 2-3
- 2.3 FIFO Buffers 2-3
  - 2.3.1 Transmit FIFO 2-3
  - 2.3.2 Transceiver FIFO 2-3
  - 2.3.3 Interrupt FIFO 2-3
  - 2.3.4 Receiver FIFO 2-3
- 2.4 Network Ring 2-4
  - 2.4.1 Protocol 2-4
- 2.5 Auxiliary Control RAM (ACR) 2-4
- 2.6 Interrupts 2-5
  - 2.6.1 Network Interrupt WRITEs 2-6
  - 2.6.2 Selected Interrupt 2-6
  - 2.6.3 Forced Interrupt 2-8
- 2.7 External Triggers 2-8
- 2.8 General Purpose Counter/Global Timer 2-8
- 2.9 LED Status Indicators 2-8
- 2.10 Modes of Operation 2-9
  - 2.10.1 Data Filter Mode 2-9
  - 2.10.2 High Performance (HIPRO) Mode 2-9
  - 2.10.3 VME Holdoff Mode 2-10
  - 2.10.4 Loopback Modes 2-10
  - 2.10.5 Write-Me-Last Mode 2-11

3. PRODUCT OVERVIEW 3-1
- 3.1 Overview 3-1
  - 3.1.1 Network Features 3-1
  - 3.1.2 Options 3-1
  - 3.1.3 VME6U Board Features 3-2
- 3.2 VMEbus Specification Level 3-2
- 3.3 Addressing Compatibility 3-3
  - 3.3.1 Memory 3-3
  - 3.3.2 Control/Status Registers 3-3
- 3.4 Data Transfer Capability 3-3
  - 3.4.1 Memory 3-3
  - 3.4.2 I/O 3-3
- 3.5 Interrupt Capability 3-3
- 3.6 P1 and P2 Connectors 3-3
- 3.7 Utility Software 3-3
  - 3.7.1 SCRAMNet Diagnostics 3-3
  - 3.7.2 EEPROM Initialization (EPI) 3-4
  - 3.7.3 SCRAMNet Monitor 3-4
- 3.8 Options 3-4

3.8.1 Electronic Bypass Switch....3-4
3.8.2 Quad Switch....3-4
3.8.3 Cabinet Kit....3-5

4. INSTALLATION....4-1
4.1 Installation Procedures....4-1
4.2 Unpack the Board....4-2
4.3 Visually Inspect the Board....4-2
4.3.1 SIMM Connections....4-2
4.3.2 Check Media Card Connection (J302)....4-2
4.3.3 J8/P2 Connection....4-4
4.4 Externally Configure the Board....4-4
4.4.1 EEPROM....4-4
4.4.2 CSR Address Switches....4-5
4.4.3 Resolution Bus Switch (S8)....4-6
4.4.4 Software Compatibility (SW_CMPT) (J2)....4-6
4.4.5 Memory Configuration (J3)....4-6
4.4.6 Variable Length Enable (VL_EN) (J4)....4-7
4.4.7 Ground Jumper (J6)....4-8
4.4.8 External Trigger Connections....4-8
4.5 Install the Board....4-9
4.6 Select Cabling Options....4-9
4.6.1 Cabinet Kit Connection....4-9
4.6.2 Fiber-Optic Configuration....4-10
4.6.3 Coaxial Cable Configuration....4-10
4.6.4 Fiber-optic Cables....4-10
4.6.5 Fiber-optic Connection....4-11
4.7 Install Fiber Optic Bypass Switch (Optional)....4-11
4.7.1 Auxiliary Connection....4-12
4.8 Internally Configure the Board....4-13
4.8.1 Node Identification....4-14
4.8.2 Network Time-out....4-14
4.8.3 Memory....4-14
4.9 Byte Swapping....4-14
4.10 Troubleshooting....4-15
4.10.1 Other H/W Considerations....4-15
4.10.2 Customer Support....4-15

5. OPERATION....5-1
5.1 Overview....5-1
5.2 Shared Memory....5-1
5.2.1 Virtual Paging....5-1
5.2.2 Memory Considerations....5-3
5.2.3 Control/Status Registers....5-3
5.3 Initialization....5-3
5.4 Basic Send/Receive Configuration....5-4
5.5 Network Ring....5-4
5.5.1 Message Contents....5-4
5.5.2 Protocol....5-5
5.5.3 Performance....5-6
5.5.4 Throughput....5-7
5.6 Auxiliary Control RAM....5-7
5.7 Interrupt Controls....5-9
5.7.1 Interrupt Options....5-9
5.8 Interrupt Conditions....5-10
5.8.1 Network Data WRITE....5-10
5.8.2 Network Error....5-14
5.8.3 Interrupt Handling....5-14
5.9 External Triggers....5-15
5.10 General Purpose Counter/Timer....5-16
5.10.1 Available Modes....5-16
5.10.2 Rollover/Reset....5-17
5.10.3 Presetting Values....5-17
5.11 Modes of Operation....5-17

5.11.1 Data Filter 5-17
5.11.2 HIPRO Mode 5-19
5.11.3 Loopback Modes 5-20
5.11.4 Node Insert Mode 5-25
5.11.5 VME Holdoff Mode 5-26
5.11.6 Write-Me-Last Mode 5-28
5.12 Quad Switch 5-28

APPENDICES

APPENDIX A. SPECIFICATIONS A-1
APPENDIX B. CSR DEFINITIONS B-1
APPENDIX C. CSR SUMMARY C-1
APPENDIX D. HOST ACCESS TIMING D-1
APPENDIX E. CABINET KIT E-1
APPENDIX F. CONFIGURATION AIDS F-1
GLOSSARY GLOSSARY-1
INDEX INDEX-1

FIGURES

Figure 2-1 Functional Diagram....2-2
Figure 2-2 ACR/Memory Access....2-5
Figure 2-3 Outgoing Interrupt....2-7
Figure 2-4 Incoming Interrupt....2-7
Figure 3-5 VME6U Board, Version B1....3-2
Figure 3-6 Node Inclusion and Isolation....3-5
Figure 4-1 Fiber-optic Media Card (Bottom view)....4-2
Figure 4-2 VME6U Layout....4-3
Figure 4-3 EEPROM WRITE (J303)....4-4
Figure 4-4 EEPROM READ (J304)....4-4
Figure 4-5 Software Compatibility (J2)....4-6
Figure 4-6 Memory Configuration (J3)....4-6
Figure 4-7 Install SIMMS....4-7
Figure 4-8 Variable Length Messages (J4)....4-7
Figure 4-9 Ground (J6)....4-8
Figure 4-10 External Trigger Connections (J7)....4-8
Figure 4-11 Fiber-optic ST Connector....4-10
Figure 4-12 Fiber-optic Connections....4-11
Figure 4-13 Inserted State (Power On)....4-11
Figure 4-14 Bypass State (Power Off)....4-12
Figure 4-15 Auxiliary Connection....4-12
Figure 5-1 Memory Sharing With Virtual Paging....5-2
Figure 5-2 Transmit Interrupt Logic....5-11
Figure 5-3 Receive Interrupt Logic....5-13
Figure 5-4 Data Filter Logic....5-18
Figure 5-5 Monitor and Bypass Mode....5-21
Figure 5-6 Wire Loopback Mode....5-22
Figure 5-7 Mechanical Switch Loopback Mode....5-23
Figure 5-8 Fiber-optic Loopback Mode....5-24
Figure 5-9 Insert Mode....5-26
Figure 5-10 Quad Switch....5-27
Figure 5-11 Interrupt Service Routine....5-29

TABLES

Table 4-1 Setting the CSR Physical Address 4-5
Table 4-2 Example of a CSR Address 4-5
Table 4-3 Setting the Resolution Switch 4-6
Table 4-4 Trigger Pin Connections (J7) 4-8
Table 4-5 External Trigger Actions 4-8
Table 4-6 Auxiliary Connection Pinout 4-12
Table 4-7 EEPROM Table 4-13
Table 4-8 EEPROM Initialization 4-13
Table 4-9 Big Endian - Little Endian Comparisons 4-14
Table 5-1 SCRAMNet+ Message Contents 5-4
Table 5-2 ACR Functions 5-8
Table 5-3 Interrupt Controls 5-9
Table 5-4 Interrupt Error/Status Conditions 5-14
Table 5-5 General Purpose Counter/Timer Modes 5-16
Table 5-6 Data Filter Options 5-17
Table 5-7 Monitor and Bypass Mode States 5-21
Table 5-8 Wire Loopback Mode States 5-22
Table 5-9 Mechanical Switch Loopback Mode States 5-23
Table 5-10 Fiber-optic Loopback Mode States 5-24
Table 5-11 Node Insert Mode 5-25

This page intentionally left blank

1. INTRODUCTION

1.1 How To Use This Manual

1.1.1 Purpose

This document is a reference manual for the SCRAMNet+ SC150 VME6U host interface board. It provides a physical and functional description of the SCRAMNet+ SC150 VME6U board. The manual describes how to unpack, set up, install and operate the hardware.

1.1.2 Scope

This information is intended for systems designers, engineers and network installation personnel. You need at least a systems level understanding of general computer processing, of memory and hardware operation, and of the specific host processor to effectively use this manual.

1.1.3 Style Conventions

- Hexadecimal values are written with "0x" preceding the value, For example, 0x03FF.
- Switch, signal and jumper abbreviations are in capital letters, for example, RSW1, J5, etc.
- Register bits and bit ranges are specified by the register identification followed by the bit or range of bits in brackets [ ], For example, CSR6[4], CSR3[15:0], ACR[1,2]
- Bit values are shown in single-quotes, for example, set bit 15 to '1'.
- Code and monitor screen displays of input and output are boxed and indented on a separate line. Text that represents user input is bolded. Text that the computer displays on the screen is not bolded. For example:

```
C:\>ls
file1          file2             file3
```

- Large samples of code are Courier font, at least one size less than context, and are usually on a separate page or in an appendix.

1.2 Related Information

SCRAMNet Network Cabinet Kit Hardware Reference (Doc. Nr. D-T-MR-CABKIT) – A physical and functional description of SCRAMNet+ cabinet kits, including installation.

SCRAMNet Network Programmer's Reference Guide (Doc. Nr. D-T-MR-PROGREF) – A collection of routines to assist SCRAMNet users with application development.

SCRAMNet Network Utilities User Manual (Doc. Nr. C-T-MU-UTIL) – A user's manual for the SCRAMNet Classic, SCRAMNet+, and SCRAMNet+ hardware diagnostic software, SCRAMNet+ EEPROM initialization software, and the SCRAMNet Network Monitor.

1.3 Quality Assurance

Systran Corporate policy is to provide our customers with the highest quality products and services. In addition to the physical product, the company provides documentation, sales and marketing support, hardware and software technical support, and timely product delivery. Our quality commitment begins with product concept, and continues after receipt of the purchased product.

Systran's Quality System conforms to the ISO 9001 international standard for quality systems. ISO 9001 is the model for quality assurance in design, development, production, installation and servicing. The ISO 9001 standard addresses all 20 clauses of the ISO quality system, and is the most comprehensive of the conformance standards.

Our Quality System addresses the following basic objectives:

- Achieve, maintain and continually improve the quality of our products through established design, test, and production procedures.
- Improve the quality of our operations to meet the needs of our customers, suppliers, and other stakeholders.
- Provide our employees with the tools and overall work environment to fulfill, maintain, and improve product and service quality.
- Ensure our customer and other stakeholders that only the highest quality product or service will be delivered.

The British Standards Institution (BSI), the world's largest and most respected standardization authority, assessed Systran's Quality System. BSI's Quality Assurance division certified we meet or exceed all applicable international standards, and issued Certificate of Registration, number FM 31468, on May 16, 1995. The scope of Systran's registration is: "Design, manufacture and service of high technology hardware and software computer communications products." The registration is maintained under BSI QA's bi-annual quality audit program.

Customer feedback is integral to our quality and reliability program. We encourage customers to contact us with questions, suggestions, or comments regarding any of our products or services. We guarantee professional and quick responses to your questions, comments, or problems.

1.4 Technical Support

Technical documentation is provided with all of our products. This documentation describes the technology, its performance characteristics, and includes some typical applications. It also includes comprehensive support information, designed to answer any technical questions that might arise concerning the use of this product. We also publish and distribute technical briefs and application notes that cover a wide assortment of topics. Although we try to tailor the applications to real scenarios, not all possible circumstances are covered.

Although we have attempted to make this document comprehensive, you may have specific problems or issues this document does not satisfactorily cover. Our goal is to offer a combination of products and services that provide complete, easy-to-use solutions for your application.

If you have any technical or non-technical questions or comments, contact us. Hours of operation are from 8:00 a.m. to 5:00 p.m. Eastern Standard/Daylight Time.

- Phone: (937) 252-5601 or (800) 252-5601
- E-mail: support@systran.com
- Fax: (937) 252-1349
- World Wide Web address: www.systran.com

1.5 Ordering Process

To learn more about Systran products or to place an order, please use the following contact information. Hours of operation are from 8:00 a.m. to 5:00 p.m. Eastern Standard/Daylight Time.

- Phone: (937) 252-5601 or (800) 252-5601
- E-mail: info@systran.com
- World Wide Web address: www.systran.com

This page intentionally left blank

2. SCRAMNET NETWORK

2.1 Overview

The SCRAMNet+ Network is a real-time communications network, based on a replicated, shared-memory concept. Each host processor on the network has access to its own local copy of shared memory that is updated over a high-speed, serial-ring network. The network is optimized for the high-speed transfer of data among multiple, real-time computers that are all solving portions of the same real-time problem. The SCRAMNet+ node board can automatically filter out redundant data.

2.2 Shared Memory

In its simplest form, the SCRAMNet+ Network system is designed to appear as general-purpose memory. The use of this memory depends only on the conventions and limitations imposed by the specific host computer system and operating system. On most processors, this means that the application program can use this memory in basically the same way as any other data storage area of memory. The memory cannot be used as instruction space.

The major difference between SCRAMNet+ memory and system memory is that any data written into SCRAMNet+ memory is automatically sent to the same shared-memory location in all nodes on the network. This is why it is also referred to as replicated shared memory. A good analogy is the COMMON AREA used by the FORTRAN programming language. Where the COMMON AREA makes variables available to subroutines of a program, SCRAMNet+ makes variables available to processors of a network.

The SCRAMNet+ memory size can range from either 4 KB or 128 KB on-board memory to 8 MB of expansion memory. Available options include: 512 KB, 1 MB, 2 MB, 4 MB and 8 MB. A software driver is usually not required except for interrupt handling. When a host computer WRITEs to the shared memory, the proper handshaking logic is supplied by the SCRAMNet+ node host adapter. The shared memory behaves somewhat like resident or local memory.

2.2.1 Dual Port Memory Controller

The Dual Port Memory Controller (see Figure 2-1) allows the host to READ from or WRITE to shared memory with a simultaneous network WRITE to shared memory. Unless an interrupt has been authorized for that memory address, the host is not aware the network is writing to shared memory. This is why caching must be disabled for SCRAMNet memory. If an interrupt has been authorized, the interrupt will then be sent to the host processor.

2.2.2 Control/Status Registers (CSRs)

The operation of the SCRAMNet+ board is controlled by Input/Output (I/O) CSRs. The location of the CSRs in the computer's address space is switch-selectable. In most

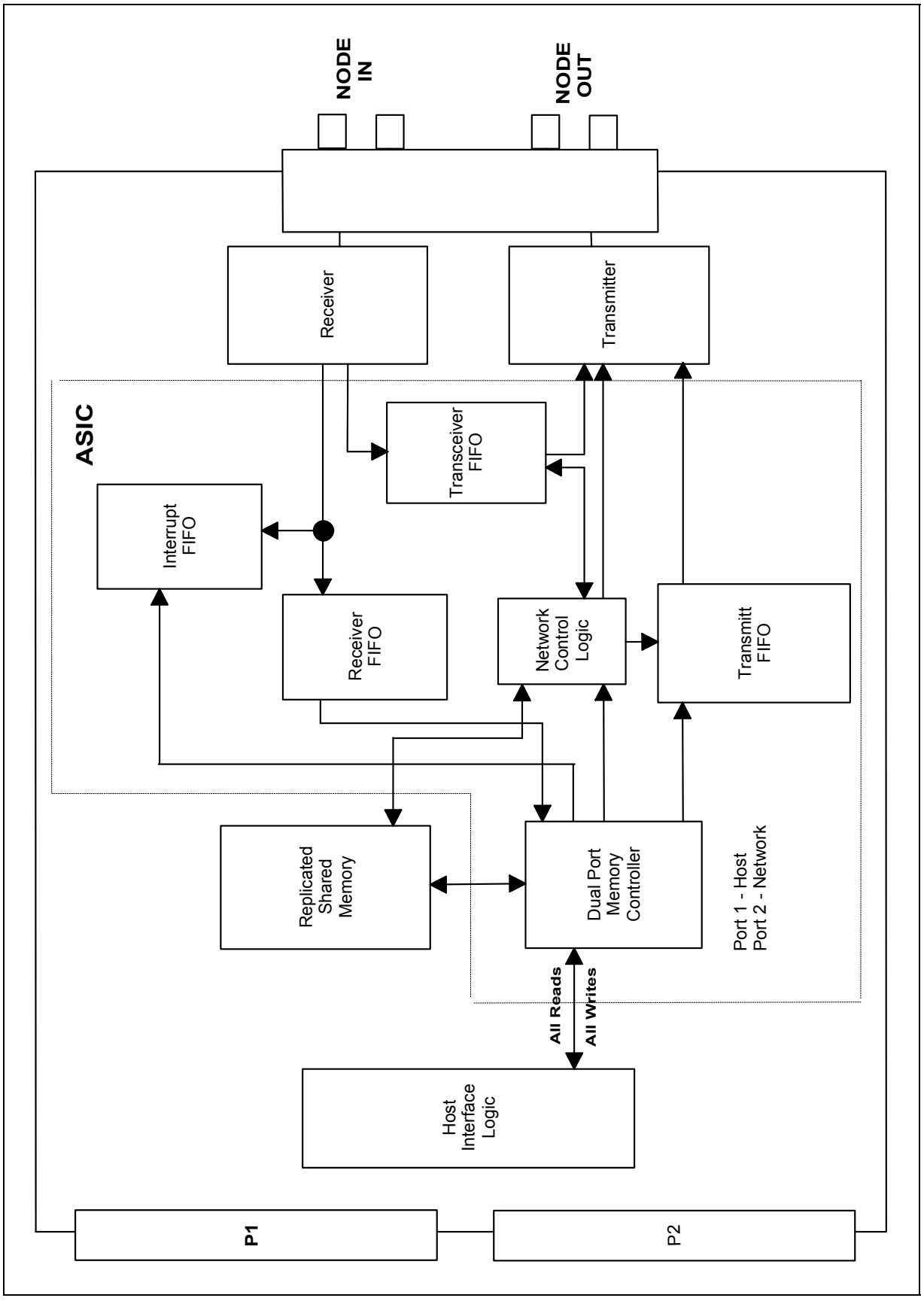


Figure 2-1 Functional Diagram

cases, the mode of operation is set during initialization and remains unchanged during run time. The CSRs are described in detail in Section 5.

2.2.3 Virtual Paging

All SCRAMNet+ nodes use the same 8 MB shared memory map. This feature permits different SCRAMNet+ boards with 4 MB of shared memory or less to be paged into different sections of the 8 MB memory map. A board with a 4 MB or smaller memory may be located on any shared memory address boundary that is an even multiple of itself (e.g., 2 MB can page to 0, 2, 4 or 6 MB address).

2.3 FIFO Buffers

The SCRAMNet+ board contains various FIFO buffers used for temporarily storing information during normal send and receive operation of the node. Refer to Figure 2-1.

2.3.1 Transmit FIFO

The Transmit FIFO is a message holding area for native messages waiting to be transmitted. Each host write to SCRAMNet+ memory may constitute a WRITE to the Transmit FIFO. (Data Filtering and HIPRO features may interfere with this.) Each WRITE to the Transmit FIFO contains 21 bits of address (A22-A2), 32 bits of data, and one bit of interrupt information. The Transmit FIFO can hold up to 1024 writes before becoming full.

When the Transmit FIFO reaches a FULL condition (CSR1[0] ON), one more host WRITE could cause a message to be lost. To prevent this, the CSR-controllable, built-in SCRAMNet+ feature called VME Holdoff extends the computer WRITE cycle until the Transmit FIFO is able to empty at least one message.

2.3.2 Transceiver FIFO

This buffer is used to receive foreign messages from the network, and send them on, or to hold received foreign messages while inserting a native message from the host onto the network.

Each node is responsible for receiving foreign messages, writing them to its copy of shared memory, and re-transmitting the message to the next node.

2.3.3 Interrupt FIFO

The Interrupt FIFO contains a 21-bit address (A22 - A2) and a retry status bit for each shared-memory-based interrupt received. The Interrupt FIFO can hold 1024 interrupt addresses. This FIFO can be read using CSR4 and CSR5.

2.3.4 Receiver FIFO

The Receiver FIFO is designed as a temporary holding place for incoming foreign messages while the shared memory is busy servicing a host request. This FIFO is three messages deep, and is designed so it can never be overrun. Each item in the Receiver FIFO contains 21 bits of address (A22 - A2), 32 bits of data, and one incoming interrupt bit. When the messages are 1024 bytes, the initial header information data stays in the FIFO, the subsequent 4 bytes of data are loaded in, and the address is incremented by four.

2.4 Network Ring

The SCRAMNet+ Network is a ring topology network. Data is transmitted at a rate of 150 Mbits/s over dual fiber-optic cables. The two lines together produce the incoming data clock. Due to the network speed and message slot size, the network can accommodate over 1,800,000 message slots passing by each node every second. There is an approximate 247 ns (minimum) delay at each node as the message slot works its way around the ring. The maximum delay depends on the selection of fixed or variable-length message packets. A fixed-length message packet has a maximum delay of 800 ns, a 256-byte variable-length message packet is 16.2 µs, and a 1024-byte variable-length message packet is 62 µs. Delay can be imposed when a node must complete the transmission of a native message packet before retransmitting a foreign message packet. A SCRAMNet+ Network can accommodate up to 256 nodes per network ring.

2.4.1 Protocol

The protocol is a register-insertion methodology and is NOT a token ring. Depending on the protocol selected, all message packets are the same size or are variable (as in the PLUS modes), and multiple nodes can transmit data simultaneously. There is no master node, and all nodes have equal priority for network bandwidth. The message protocol is designed specifically for real-time applications where data must be passed very rapidly. When the node operates in BURST or BURST PLUS mode, the node will never re-transmit its own messages for error correction. When operating in PLATINUM or PLATINUM PLUS mode, error detection is enabled, and re-transmission can occur.

BURST MODE

BURST mode is an open loop, non-error-corrected communication mode. This mode allows multiple 82-bit messages (46-bit header plus 32-bite data and four parity bits) per node on the ring at a time. The limited packet length enhances the data latency characteristics of the network by providing the shortest possible media access delay. The messages are transmitted as fast as the system will allow.

PLATINUM MODE

PLATINUM mode is BURST mode with error correction enabled. The messages are transmitted as fast as the system will allow, but error checking is used to detect and re-transmit corrupted packets.

PLUS MODES

The PLUS mode protocol enhancement can increase the maximum network throughput from 6.5 MB/sec to approximately 15.2 to 16.7 MB/sec by the use of variable-length message packets. Each SCRAMNet+ message packet has a 46-bit header plus the data. The user-selectable maximum packet size increases the data size from the normal 32 bits to either 256 or 1024 bytes of data. Data must be written to sequential longword addresses.

2.5 Auxiliary Control RAM (ACR)

The Auxiliary Control RAM (ACR) provides a method of external triggering and interrupt control by offering a choice of four actions to occur when a particular SCRAMNet+ shared-memory address is written into. Each shared-memory location has its own action or set of actions associated with it.

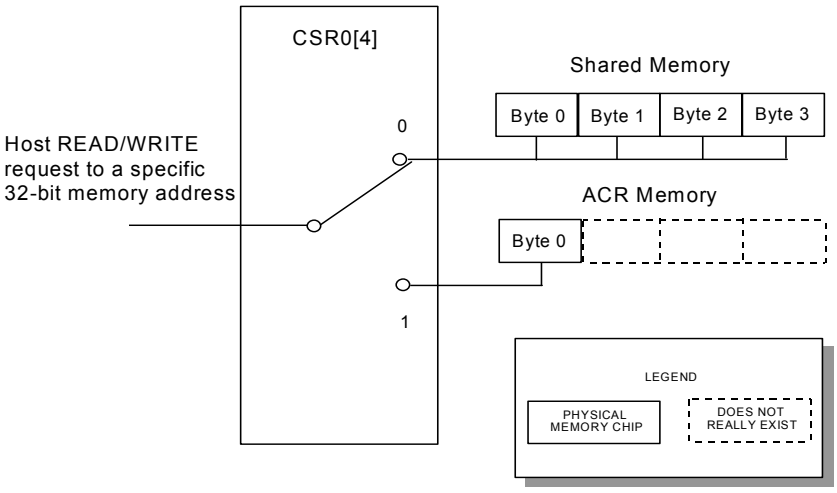


Figure 2-2 ACR/Memory Access

In Figure 2-2, host CPU READ/WRITE operations are channeled to either SCRAMNet+ memory or to the ACR. The ACR is a physically separate memory from the shared memory. Channeling is based on a user-controlled switch setting and may be toggled to the desired position by writing to a bit in the SCRAMNet+ CSR. When access to the ACR is enabled, shared memory is not accessible by the host and the ACR byte is viewed as the least significant byte (LSB) of every shared-memory four-byte address. The ACR bits define what external trigger and/or interrupt action(s) are to be taken whenever writing to any byte of the SCRAMNet+ shared memory 4-byte word.

Only five bits of the ACR are associated with every four-byte word of shared memory (on even four-byte boundaries). The other 27 bits of the ACR are phantom bits and do not physically exist.

2.6 Interrupts

SCRAMNet+ allows a node processor to receive interrupts from and transmit interrupts to any node on the network, including the originating node, provided the receiving node is set up to receive an interrupt message. Interrupts can be generated under two different conditions:

- SCRAMNet+ Network data WRITEs to shared memory; and
- SCRAMNet+ network errors detected on the local node.

SCRAMNet+ interrupts usually require a device driver to interface with the node processor. There must also be a host-dependent interrupt vector placed in CSR6 and CSR7 identifying the Interrupt Service Routine (ISR). The driver is required primarily to permit the host processor to handle interrupts from the SCRAMNet device.

2.6.1 Network Interrupt WRITEs

FOREIGN MESSAGE

The node can receive a message from another node with the interrupt bit set. If Receive Interrupt Enable ACR[0] and Interrupt Mask Match Enable CSR0[5] are enabled, the data is written to shared memory and the address is placed on the Interrupt FIFO.

NATIVE MESSAGE

If the message received was originated by the node, and Write Own Slot Enable CSR2[9] and Enable Interrupt on Own Slot CSR2[10] are enabled, the host has authorized a Self-Interrupt. The data is written to shared memory and the address is placed on the Interrupt FIFO.

Network Interrupt WRITEs can be accomplished by two methods:

- **Selected**. Data WRITEs to selected shared memory locations from the network.
- **Forced**. Any data WRITEs to any shared memory from the network.

In either case, the node can be configured to WRITE to itself. This condition is called "Self Interrupt."

2.6.2 Selected Interrupt

The selected-interrupt method requires choosing SCRAMNet+ shared-memory locations on each node to receive and/or to transmit interrupts. These shared-memory locations may also be used to generate signals to external triggers. The procedure for selecting shared-memory locations for interrupts and/or external triggers is explained in the paragraph on the Auxiliary Control RAM, paragraph 2.5.

OUTGOING INTERRUPT

The Outgoing Interrupt is described in Figure 2-3. If both Transmit Interrupt Enable ACR[1] and Network Interrupt Enable CSR0[8] are set, and a data item is transmitted to any of the selected-interrupt memory locations, then an interrupt message is sent out on the network. This message will generate interrupts to any processors on the network that have that same shared-memory location selected to receive interrupts.

INCOMING INTERRUPT

Figure 2-4 demonstrates the process of receiving a message with the interrupt bit set. The data is written to shared memory and the address is placed in CSR5 and CSR4 to await being sent to the host. If the Receive Interrupt Enable ACR[0], Host Interrupt Enable CSR0[3], and the Interrupt Memory Mask Match Enable CSR0[5] are set, and network interrupt data is received for any one of the selected-interrupt memory locations the following occurs:

- The data is stored in that location
- The SCRAMNet+ address of the memory location is placed on the Interrupt FIFO queue, and
- An interrupt is sent to the processor.

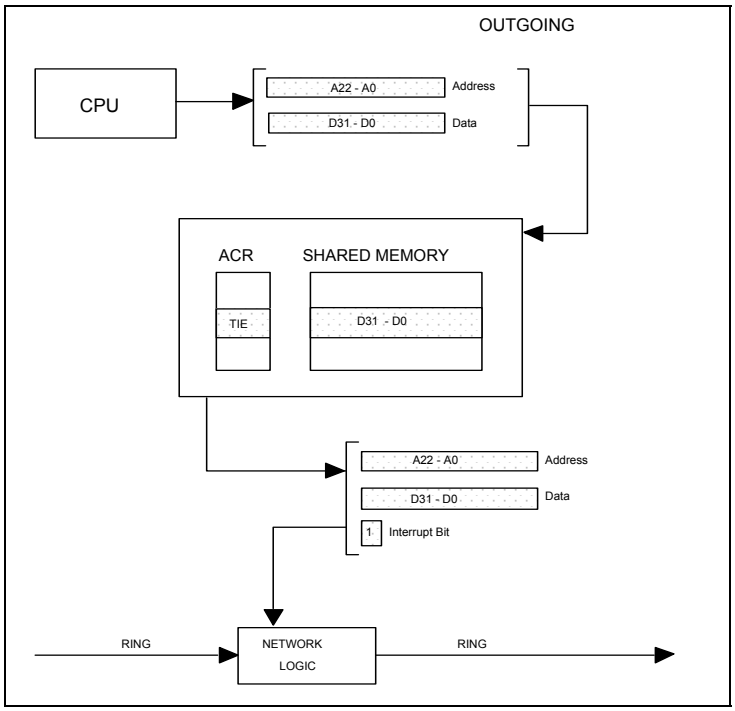


Figure 2-3 Outgoing Interrupt

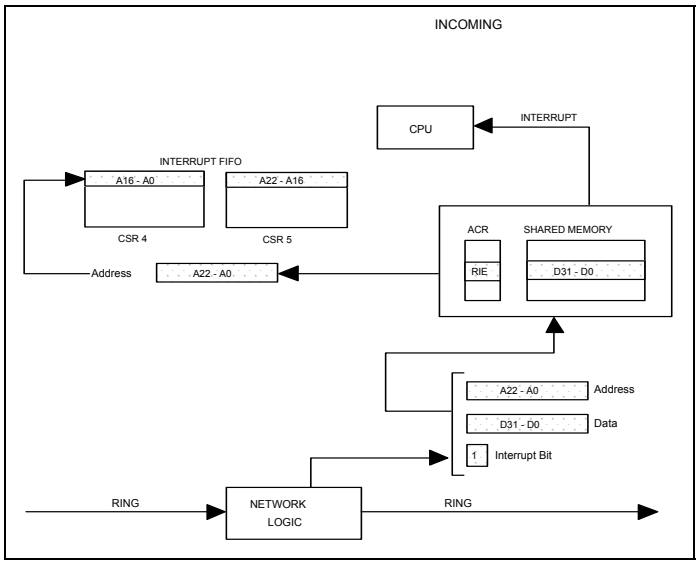


Figure 2-4 Incoming Interrupt

NETWORK ERRORS

The Interrupt on (Network) Errors mode is enabled by setting CSR0[7] ON. Network errors are defined in CSR1 according to an interrupt mask set in CSR9. When an incoming foreign message generates an interrupt, there is no way to mask the interrupt according to the content of the message. However, specific error conditions may be identified.

Error conditions are listed in CSR1 and may be masked by setting the corresponding bit in CSR9. If the Mask bits in CSR9 are all set to '1,' any error will generate an interrupt. Otherwise, only errors with a '1' in the appropriate Mask bit will generate an interrupt.

A Network Interrupt vector may be placed in CSR7 to identify a Network Error Interrupt Service Routine.

2.6.3 Forced Interrupt

The forced-interrupt method works the same as for selected except for the choice of interrupt locations. All shared-memory locations are automatically set up to receive and/or transmit interrupts depending upon the ACR override conditions set in CSR0[6] and/or CSR0[9].

When Override Receive Interrupt Enable CSR0[6] is set an interrupt will be sent to the host by any network interrupt data message, regardless of the status of the ACR Receive Interrupt bit.

When Override Transmit Interrupt Enable CSR0[9] is set an interrupt will be sent out on the network regardless of the status of the ACR Transmit Interrupt bit.

A third condition, Receive Interrupt Override CSR8[10], is used to designate all incoming network traffic as interrupt messages. The network message interrupt bit does not need to be set.

2.7 External Triggers

Two external triggers are supported by SCRAMNet+ VME6U. The external triggers will occur only if the ACR has been configured to enable them. Triggers 1 and 2 are generated by SCRAMNet+ shared-memory access. Both triggers generate a 26.64 ns TTL level compatible, non-terminated, output.

- **Trigger 1** - Host Read/Write ACR[2] enables)
- **Trigger 2** - Network Write ACR[3] enables)

2.8 General Purpose Counter/Global Timer

The General Purpose Counter/Timer has six modes of operation controlled by CSR8 and CSR9, the output from the General Purpose Counter/Timer is stored in CSR13. Counter modes can count errors, external trigger events, or network messages. A high-resolution timer mode can run free or measure the ring time with a 26.66 ns resolution.

The global timer mode clocks with a resolution of 1.706 µs and resets on an external trigger event. (See 2.7: External Triggers). A specific shared-memory location may be identified with External Trigger 2 (ACR[3]) so that a memory WRITE from a single node on the network can simultaneously reset all the global timers in the ring.

If the Trigger 2 event is the frame counter, the timers in the ring effectively become synchronized sub-frame timers, which can then be used to tag time-critical data or to measure and compare the completion time of various tasks within a distributed real-time system.

2.9 LED Status Indicators

INSERT

The green Insert LED is ON when the node is inserted into the SCRAMNet+ Network ring. This is the result of setting CSR 0[15].

MESSAGE WAITING

The green Message Waiting LED lights when a message is placed in the Transmit FIFO.

CARRIER DETECT

The green carrier detect LED is ON when there is a valid pair of transmit lights from the previous SCRAMNet+ node into this node's receiver pair. If the fiber-optic cables are connected and the carrier detect LED is OFF, then the ring integrity is NOT valid. This condition indicates improper fiber-optic cabling or problems with the down-line node's transmitter(s).



NOTE: On a freshly powered system, a message from any node on the ring may be necessary to establish carrier.

ERROR

The yellow error LED is active whenever any error conditions set in CSR1 are detected.

NATIVE MESSAGE

The green Native Message LED lights when the message received was originated by the node.

FOREIGN MESSAGE

The green Foreign Message LED becomes active when the message received is from another node.

2.10 Modes of Operation

2.10.1 Data Filter Mode

When SCRAMNet+ Data Filtering is enabled, only those WRITEs to SCRAMNet+ memory that produce a data change are transmitted to the network.

EXAMPLE:

If location 1000 in SCRAMNet+ memory contains the value '20' and the host processor WRITEs the value '20' to location 1000, then no network traffic will be generated. However, if any other value is written to location 1000, then the new value will be passed around the network to update the other SCRAMNet+ node memories.

When a WRITE is received from the host, a comparison is made to the old data at that address to see if there was a change before writing to shared memory. If the data has changed, then it is written to shared memory, and is also transmitted onto the network. This entire process is completed within the host memory standard bus WRITE cycle.

Data filtering is a powerful communications compression technique for cyclical applications. This technique has been shown to significantly reduce the network traffic and therefore increase the effective throughput on the network.

2.10.2 High Performance (HIPRO) Mode

HIPRO provides an efficient means to transmit 8-bit and 16-bit data transactions as one 32-bit network WRITE. It also provides a means of keeping 32-bit data from becoming fractured.

EXAMPLE #1:

A floating-point length numeric sent in 8-bit or 16-bit pieces may not be accurately re-assembled at the destination.

EXAMPLE #2:

The receiving node may otherwise try to use part or half of such a value before the entire 32 bits is received.

HIPRO WRITE

The SCRAMNet+ network message is based on 32-bit longword data. This means if any 8-bit field of the 32-bit buffer is changed, the entire 32-bit message is transmitted. If a host is limited to only 8-bit or 16-bit databus transactions the network throughput is quartered or halved, respectively.

HIPRO mode permits a 32-bit location to be set up in shared memory such that any initial WRITE smaller than 32 bits to that location will not automatically go onto the network. The 32-bit WRITE to the network will only occur when all four bytes within the 32-bit location have been written through subsequent WRITEs by the host CPU. This can be accomplished by four consecutive 8-bit or two consecutive 16-bit WRITEs to the SCRAMNet memory.



NOTE: HIPRO WRITE will not work if Disable Host to Memory Write CSR2[8] is set, or when writing two separate shortwords while using interrupts.

HIPRO READ

The HIPRO READ is controlled by CSR16. This register is CSR enabled and ACR location selectable.

To conserve host cycles and increase host throughput, HIPRO READ mode allows the host to get part of the information (1 shortword or 2 bytes) during the first READ on that longword boundary. On the next READ operation (not the same location or within the same longword boundary) the remaining data is provided.

2.10.3 VME Holdoff Mode

It is possible that the Transmit FIFO can become full when the host is writing to the SCRAMNet+ interface faster than the network can absorb the data.

In VME Holdoff mode, the host WRITE cycle is automatically extended until the SCRAMNet+ Transmit FIFO buffer transmits at least one message. This prevents the loss of data and is transparent to the user.

In some system designs, and on some computer buses, it is not desirable or effective to have the WRITE cycle lengthened to match network throughput—even at the expense of possible data loss across the network. In this case this option may be disabled by setting CSR8[1] ON. Transmit FIFO 7/8 Full CSR1[2] can then be used to control the data flow via software control.

2.10.4 Loopback Modes

Loopback mode is used for testing, and for routing data, which would normally be transmitted onto the network back into the node. This mode is used to check performance internally (Wire Loopback) at the Media Card (Mechanical Switch Loopback) and Transmit/ Receive (Fiber-optic Loopback).

WIRE LOOPBACK MODE

The Wire Loopback mode needs no manual external modifications to work. Wire Loopback is enabled by setting CSR2[7] ON. This mode checks the on-board circuitry for continuity.



NOTE: If a node is inserted into the network while in wire loopback mode, it will create a break in the network ring, making all nodes down-line unreachable.

MECHANICAL SWITCH (MEDIA CARD) LOOPBACK MODE

Mechanical Switch (Media Card) Loopback mode is enabled by setting Mechanical Switch Override CSR8[11] to OFF. This test is used to check the circuitry up to and including a major portion of the Media Card but excludes the fiber-optic circuitry. In this test the signal does not leave the Media Card.

FIBER-OPTIC LOOPBACK MODE

The Fiber-optic Loopback mode must have the optional Fiber Optic Bypass Switch connected, Disable Fiber-optics Loopback CSR2[6] set to OFF (power up default), and Insert Node CSR0[15] enabled to be valid. When the Fiber-optic Loopback mode is in effect, the output of the transmitter is connected by fiber optics directly to the input of the receiver, and the receiver is disconnected from the network.

The optional Fiber Optic Bypass Switch must be installed for this loopback to work. However, in the absence of the Fiber Optic Bypass Switch, fiber-optic cables could be run from the node's transmitter output connectors to the receiver input connectors. This configuration, with Insert Node enabled, would constitute a Fiber-optic Loopback mode for stand-alone testing. Set CSR2[6] ON to disable the Fiber-optic Loopback mode when the node is in use as a part of the network. This configuration is not a substitute for the Fiber Optic Bypass Switch for network operation.

2.10.5 Write-Me-Last Mode

The Write-Me-Last mode of operation allows the originating node to be the last node in the ring to have the data deposited to its memory. This can be useful for synchronization. This means that when the host performs a WRITE to the SCRAMNet+ shared memory, this data is not immediately written to the host node's memory, but is first sent to the other nodes on the network. When the message returns to the originating node it is written to shared memory, and is then removed from the network ring.

Therefore, host-originated data written to shared memory travels the ring updating the SCRAMNet+ node memories on the ring and, upon returning to the originating node, that node WRITEs the data to its own shared memory as the last node on the ring. This guarantees that the data is available on all other nodes.

This page intentionally left blank

3. PRODUCT OVERVIEW

3.1 Overview

SCRAMNet+ (Shared Common Random Access Memory Network) is a communications network geared toward real-time applications, and based on a replicated, shared-memory concept.

The SCRAMNet+ VME6U host interface node board is backwards compatible with the original SCRAMNet Classic product. The SCRAMNet Classic Gold Ring communication protocol is compatible with the SCRAMNet+ Platinum protocol, but not with BURST PLUS or PLATINUM PLUS. The programmable byte swapper is no longer available on VMEbus-based products.

The SCRAMNet+ VME6U board requires a single slot in the VMEbus chassis.

The SCRAMNet+ VME6U board base address for Control/Status Registers (CSRs) is switch selectable. The 4 KB or the optional 128 KB on-board shared memory can be upgraded to 512 KB, 1 MB, 2 MB, 4 MB or 8 MB random access memory (RAM). Installing any memory upgrade overrides the on-board 128 KB memory.

3.1.1 Network Features

- A ring topology with 150 Mbit/s line transmission rate.
- A Data Filter that allows only data stored in shared memory that has changed to be communicated to the other network nodes.
- Field Upgrade Memory Options up to 8 MB of replicated, shared memory for each node processor.
- BURST Mode protocol (Error Correction Disabled) with fixed-length message packets of 82 bits.
- BURST PLUS Mode communication based on variable-length message packet to a maximum of either 256 bytes or 1024 bytes.
- PLATINUM Mode protocol (Error Correction enabled) with fixed-length message packets of 82 bits.
- PLATINUM PLUS Mode communication based on variable-length message packet to a maximum of either 256 bytes or 1024 bytes.
- 256 node capacity on each ring.
- No operating or system software required to support network protocol.
- No network-dependent application software required.

3.1.2 Options

- Optional paired-fiber-optic or coaxial transmission media
- Fiber Optic Bypass Switch for ring continuity when node power is off.
- Quad Switch—A switching-control device that controls up to four nodes or sub-rings, eliminates the need for a separate Fiber Optic Bypass Switch, and functions as a repeater.

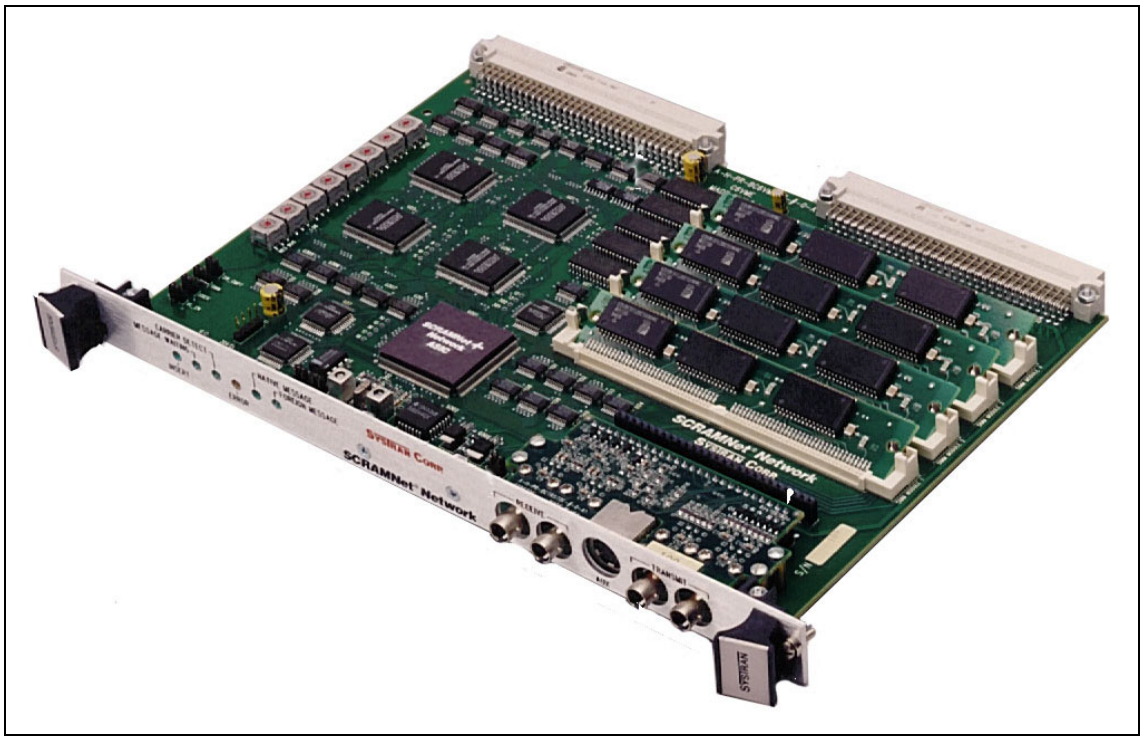


Figure 3-5 VME6U Board, Version B1

3.1.3 VME6U Board Features

- SIMM memory upgrade option
- General purpose counter
- Error-interrupt mask
- Dynamic shared-memory addressing
- Switch-selectable CSR address selector
- Virtual paging for shared memory (CSR selectable)
- Variable-length message packet capability
- Dual-port memory
- Dual-vector memory/error interrupt (single level interrupt)
- Single-slot solution
- EEPROM initialization

3.2 VMEbus Specification Level

The SCRAMNet+ VME6U host board was designed in accordance with the VMEbus specification Revision C.3, ANSI/IEEE Std 1014-1987.

- Slave device
- SADO32 (No UAT, no BLT; A32, A24 memory; A32, A24, A16 CSR)
- SRMW32 (D32, D16, D08 (EO))
- 8-bit vector ROAK
- 6U double-height card size

3.3 Addressing Compatibility

3.3.1 Memory

The shared memory resident on the SCRAMNet+ VME6U host interface board must be located on either the A24 standard bus or the A32 extended bus. The address is CSR selectable. The memory address selected must be an address boundary that is a multiple of the shared-memory size, and must be loaded and enabled through the CSRs.

3.3.2 Control/Status Registers

I/O Control in the form of CSRs can be located on the A16 I/O bus, the A24 standard bus, or the A32 extended bus. The address is set manually using the rotary switches on the board. This is independent of the shared-memory address. The CSRs require 64 contiguous bytes of address space.

3.4 Data Transfer Capability

3.4.1 Memory

Data transfers to the shared memory on the SCRAMNet+ VME6U host interface can be 8, 16 or 32 bits wide and may be of the READ-MODIFY-WRITE type. Three-byte unaligned transfers are not permitted.

3.4.2 I/O

Data transfers to the I/O control area of the SCRAMNet+ VME6U host interface can be 8, 16 or 32 bits wide and may be of the READ-MODIFY-WRITE type. Three-byte unaligned transfers are not permitted.

3.5 Interrupt Capability

The SCRAMNet+ VME6U host interface is an interrupter of the type D08. Interrupt level 1 through 7 may be selected by setting a value in CSR15. The IRQ is mapped using a bit-wise format; bit 1 is IRQ 1, bit 2 is IRQ 2, etc. (Bit 0 is not used.) The vector generated by the node is 8 bits wide. Being an ROAK (Release On interrupt AcKnowledge) type of interrupter means that the device releases the interrupt request during the interrupt-acknowledge cycle. The 8-bit vector address is loaded at CSR6 - Memory, and CSR7 – Error. If Interrupt-on-Error is not used, CSR7 must contain the same vector as CSR6.

3.6 P1 and P2 Connectors

The SCRAMNet+ VME6U card's P1 and P2 backplane connectors are in accordance with the VMEbus specifications.

P2 pin connections are defined for SCRAMNet+ in Appendix A.

3.7 Utility Software

3.7.1 SCRAMNet Diagnostics

The SCRAMNet Network Hardware Diagnostics are designed to test the functionality of the hardware. This suite of tests will detect whether it is testing a Classic board or a SCRAMNet-LX/SCRAMNet+ board and adjust the test menus accordingly.

3.7.2 EEPROM Initialization (EPI)

The EEPROM Initialization program is a SCRAMNet+ utility used to simplify configuration of the network node. The EPI program will store a start-up configuration in the serial EEPROM that can initialize the node on power up. This initialization program can be run when the board is installed to set the desired power-up state of the SCRAMNet+ node. EPI is completely menu driven and contains a context-sensitive help feature.

3.7.3 SCRAMNet Monitor

The SCRAMNet Monitor allows viewing and editing of memory and CSR locations on the SCRAMNet node. This utility is useful during software development to verify that the correct values are being written to SCRAMNet memory and CSRs.

3.8 Options

3.8.1 Electronic Bypass Switch

The Electronic Bypass Switch exists on some Media Cards. This switch allows for fast bypass on power-fail conditions. The electronic switch operates in the low nanosecond range compared to a 20 millisecond time for a typical mechanical switch.

In case of node power failure, the electronic switch restores the network so quickly that only one or two messages will have to be retransmitted, whereas a mechanical switch could cost an excessive amount of transmission time re-sending perhaps thousands of messages.

3.8.2 Quad Switch

The SCRAMNet Quad Switch is designed to provide configuration control over the network topology and computing resources. The Quad Switch allows local clusters of up to four SCRAMNet nodes to be switched in or out of a primary SCRAMNet ring, independently and dynamically (Figure 3-6). It also allows sharing of a critical real-time resource between multiple systems.

The Quad Switch performs other useful functions such as optical bypassing, fiber-optic repeating to gain transmission length beyond the SCRAMNet node's transmission power limit, and to act as a media converter.

The electronic bypass switching action is very fast, introducing a total network disruption of about one microsecond. This is over 10,000 times faster than mechanical optical bypass switches, and permits ring re-configuration to be performed in real-time with minimal impact on the system.

As a repeater, each Quad Switch port converts optical signals to electrical signals. These signals are re-synchronized and re-transmitted. This allows each connection to the Quad Switch to be the maximum length for the type of media selected.

The Quad Switch can also perform media conversion. Since each port has a Media Card just like a SCRAMNet node, each port can be configured to handle coax, standard link or long-link fiber. This allows a signal to arrive on one media type, and go out on another.

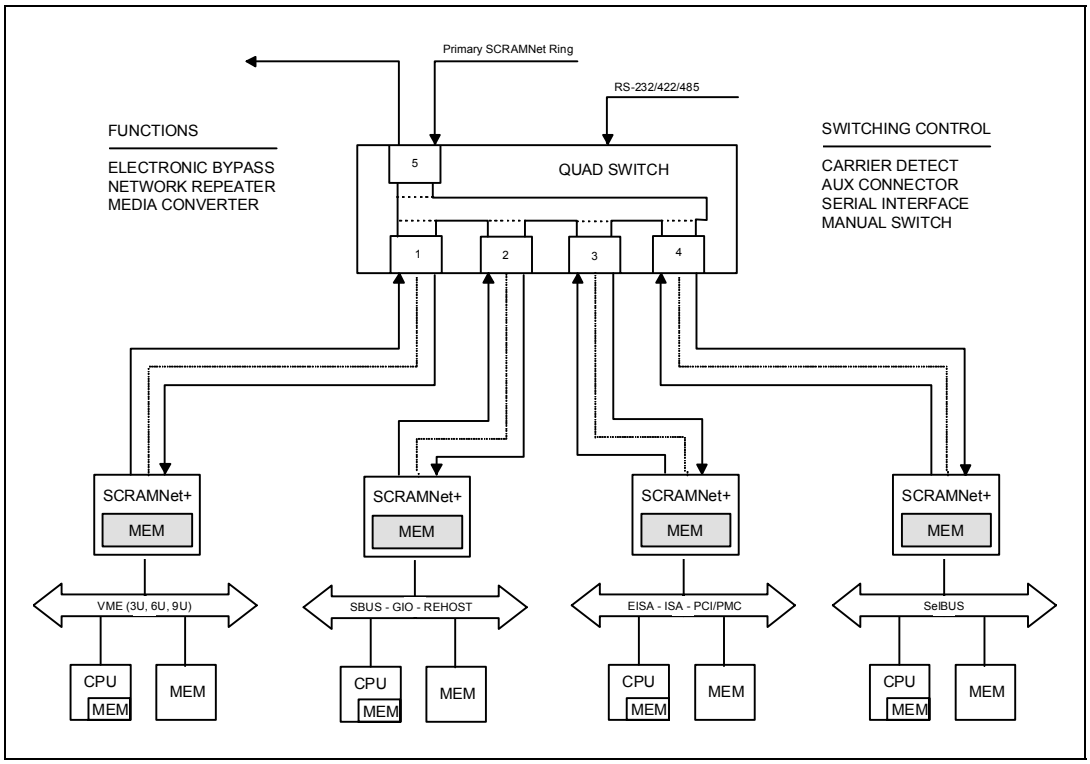


Figure 3-6 Node Inclusion and Isolation

Visually, LED's signify the state of node inclusion in the ring and if carrier is detected. If carrier is not detected, the port is put into Isolate state and the port is bypassed thereby retaining ring integrity. The auxiliary connector and the associated control cable links the port to the node to allow the application running the node to switch the Quad Switch in and out of Include or Isolate state. A manual Include/Isolate switch is used to guarantee that a node is isolated or that it can be included. A serial-port interface is used to send message packets to the Quad Switch to perform control function or to obtain switch status remotely via the RS-232 or RS-422/485 connection. Two mechanical rotary switches set the serial interface address.

3.8.3 Cabinet Kit

The Cabinet Kit for the SCRAMNet+ Network permits adapting the node to the host cabinet while still maintaining the shielding of the chassis. Access to the node's connections, and cable integrity is exterior to the cabinet. This eliminates the need to remove the node board, once proper installation has been accomplished.

Cabinet kits are available for a variety of vendor chassis, and include the Compact and the Extended models. The Compact Cabinet Kit is described in Appendix D, and in detail in the *SCRAMNet Network Cabinet Kit Hardware Reference*.

This page intentionally left blank

4. INSTALLATION

4.1 Installation Procedures

Installation of the VME6U board includes the following:

- Unpack the board
- Visually inspect the board
  - Check SIMM connection(s), if any
  - Check Media Card connection
- Externally configure the board
  - Set/verify EEPROM WRITE jumper (J303)
  - Set/verify EEPROM READ jumper (J304)
  - Set CSR Address switches (S1-S7)
  - Set bus resolution switch (S8)
  - Set/verify Software Compatibility jumper (J2)
  - Set/verify Memory Configuration jumper (J3)
  - Set/verify Variable Length jumper (J4)
  - Set/verify Ground jumper (J6)
  - Set/verify External Trigger connections (J7)
- Install the board
- Select cabling options
- Install Fiber Optic Bypass Switch (optional)
- Internal Configuration
  - Set Node Identification (CSR3, BITS [15:8])
  - Set Network Time-out (CSR5)
  - Set Memory Base Address (CSR10, bits [15:12] and CSR11, bits [15:0])
  - Enable/disable Shared Memory (CSR10, bit 0)
  - Set Variable Length message packet size (CSR2, bit 11)

4.2 Unpack the Board

1. The board is wrapped in an anti-static bag and encased in anti-static foam.



CAUTION: Use an anti-static mat connected to a wristband when handling or installing the SCRAMNet+ board.

2. Remove the board and anti-static bag from the carton. Open the anti-static bag and remove the board while wearing an anti-static wristband.

Save the shipping material in the event that the SCRAMNet+ board needs to be returned.

The optional fiber-optic cables, the optional Fiber Optic Bypass Switch, and the optional redundant transceivers are shipped in separate cartons.

4.3 Visually Inspect the Board

The SCRAMNet+ Network node consists of a single board as shown in Figure 4-2. If the optional memory upgrade was ordered, it will come already installed. In the event that any shipping damage has occurred, notify SYSTRAN Corporation or your supplier immediately.

4.3.1 SIMM Connections

If SIMMs are installed, press gently downward on each SIMM and make sure the clips are engaged to hold the SIMM in place.

4.3.2 Check Media Card Connection (J302)

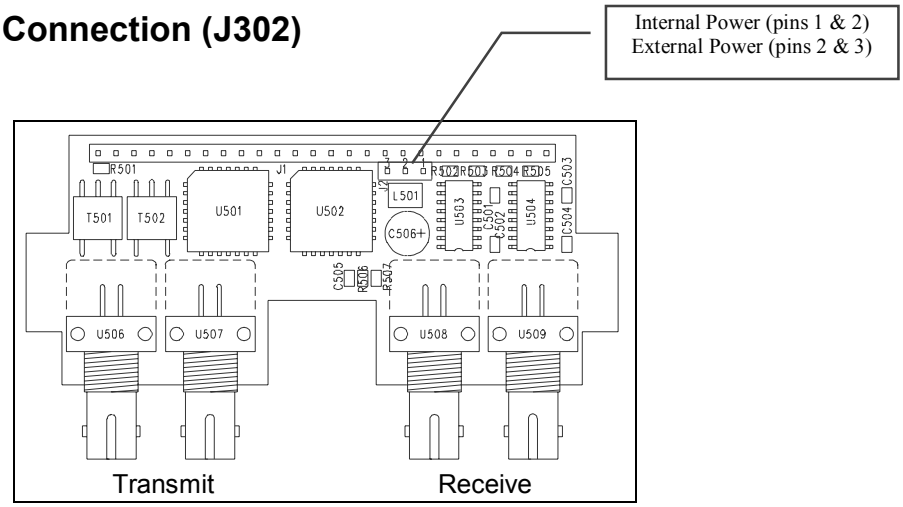


Figure 4-1 Fiber-optic Media Card (Bottom view)

The Media Card can have either coaxial or fiber-optic connectors. Figure 4-1 shows the fiber-optic media card. The SCRAMNet+ VME6U board will support either option. The type of Media Card installed will be dictated by the network configuration. There are two receive connections (Rx_1 and Rx_2) and two transmit connections (Tx_1 and Tx_2).

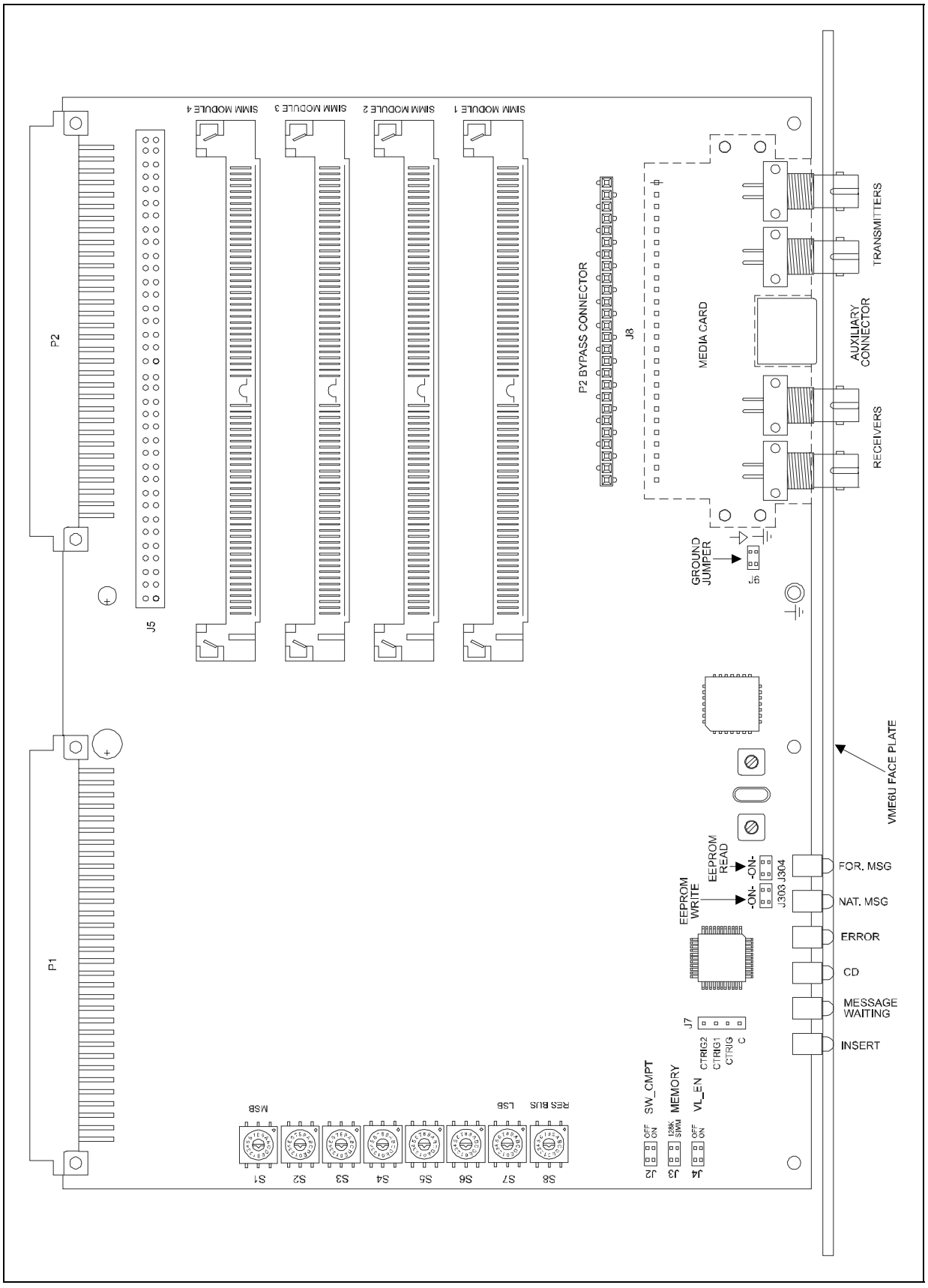


Figure 4-2 VME6U Layout

This fiber-optic card has two power options; host power and standby or battery power. Jumper J2 in Figure 4-1 controls the power options. Pins 1 and 2 are for normal host power, and pins 2 and 3 are for standby power. The standby or battery power requires external connection via the auxiliary connection on the cabinet kit board or the host interface board if no cabinet kit is installed.

4.3.3 J8/P2 Connection

If a cabinet kit is used in lieu of the Media Card, an adapter card is installed on J302 and J8. This forces the Media Card signals to J5. When four 9-pin headers are installed on J5, the Media Card Signals pass through to P2, row A. See Appendix C for the pinout description. Connection J8[26:1] corresponds to connection P2[26:1]. Refer to the SCRAMNet *Network Cabinet Kit Hardware Reference* for more details.

4.4 Externally Configure the Board

4.4.1 EEPROM

EEPROM WRITE JUMPER (J303)

To enable the EEPROM WRITE place a 2-pin header on the top row as viewed from the face-plate edge of the board. To disable EEPROM WRITE place a 2-pin header on the bottom row. (Figure 4-3). Factory default setting: ENABLED

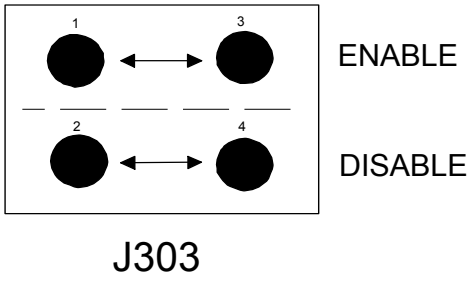


Figure 4-3 EEPROM WRITE (J303)

EEPROM READ JUMPER (J304)

To enable the EEPROM READ place a 2-pin header on the top row as viewed from the faceplate edge of the board. To disable EEPROM READ place a 2-pin header on the bottom row. (Figure 4-4). Factory default setting: ENABLED

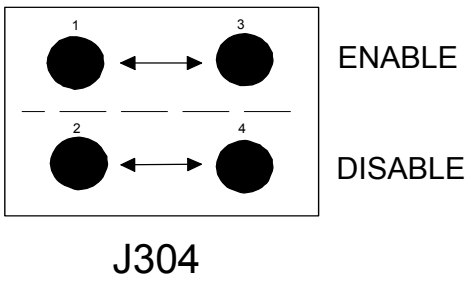


Figure 4-4 EEPROM READ (J304)

4.4.2 CSR Address Switches

The CSR addressing for the VME6U bus can be short (16-bit), standard (24-bit) or extended (32-bit). The SCRAMNet+ CSRs must be configured on a contiguous 64-byte boundary.

Set the base address for the Control/Status Registers using S1 through S7 as indicated in Table 4-1 and Figure 4-2. Switch S1 contains the Most Significant Bits, and S7 contains the Least Significant Bits.

Factory Default settings: CSR ADDRESS: S1-S7 = 00000000

Table 4-1 Setting the CSR Physical Address

Switch	Physical Address			
S1	A31	A30	A29	A28
S2	A27	A26	A25	A24
S3	A23	A22	A21	A20
S4	A19	A18	A17	A16
S5	A15	A14	A13	A12
S6	A11	A10	A09	A08
S7	A07	A06	Not used	Not used

Table 4-2 Example of a CSR Address

S1	S2	S3	S4
A31 A30 A29 A28	A27 A26 A25 A24	A23 A22 A21 A20	A19 A18 A17 A16
1	A	B	C

S5	S6	S7	—
A15 A14 A13 A12	A11 A10 A09 A08	A07 A06 ---- ----	--- ---- ---- ----
3	0	0	0

Table 4-2 contains an example of CSR address 0x1abc3000.



NOTE: The least significant hex digit is always zero, and no rotary switch is provided for bits. A[3:0]. Bite A15 and A4 are also always zero on switch S7. Switch S7 can only have the values 0, 4, 8 or C.

EXAMPLE of the 16-bit address 0x6000 = :

S5	S6	S7
6	0	0

4.4.3 Resolution Bus Switch (S8)

Switch S8 is the Resolution Bus switch. Set switch S8 to the desired value as indicated in Table 4-3 and Figure 4-2.

Resolution bus switch factory default setting: 0x6-CSR A16 MEMORY A24

Table 4-3 Setting the Resolution Switch

CSR	Memory	Value
A32	A32	F
A32	A24	E
A24	A32	B
A24	A24	A
A16	A32	7
A16	A24	6

4.4.4 Software Compatibility (SW_CMPT) (J2)

If the SCRAMNet+ board must be software compatible with SCRAMNet Classic interrupt addressing, set place a 2-pin header on the bottom row as viewed from the face-plate edge of the board (Figure 4-5).

Factory default setting: OFF

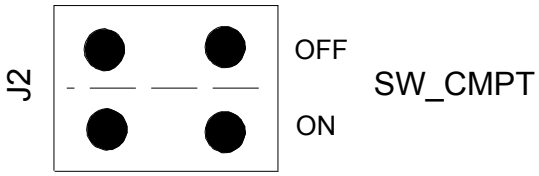


Figure 4-5 Software Compatibility (J2)

4.4.5 Memory Configuration (J3)

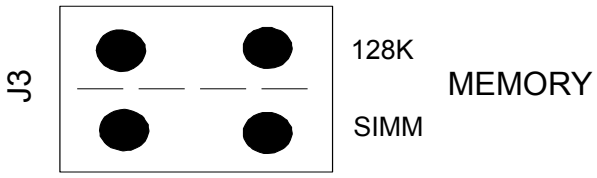


Figure 4-6 Memory Configuration (J3)

If no SIMMs are installed, two options are available

- 4 KB memory—install a 2-pin header on the bottom row.
- 128 KB memory—install a 2-pin header on the top row.

If SIMMs are installed, install a 2-pin header on the bottom row.

The SIMMs used with this board are proprietary, and must be ordered from Systran.

OPTIONS

Low density, 512 KB SIMMs 1 = 512 KB (SIMM 1)
2 = 1 MB (SIMM 1 and 2)
4 = 2 MB (SIMMs 1 through 4)

High density, 2 MB SIMMs 1 = 2 MB (SIMM 1)
2 = 4 MB (SIMM 1 and 2)
4 = 8 MB (SIMMs 1 through 4)

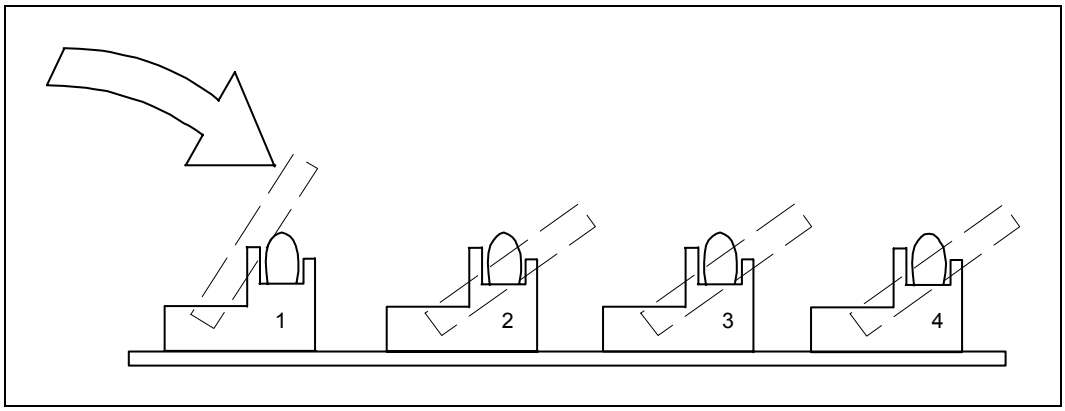


Figure 4-7 Install SIMMS

SIMMs must all be either low density or high density; they can not be mixed.

To install SIMMs, set the SIMM in the slot and gently press back and down until the clips snap into place (Figure 3-6).

To remove the SIMMs, push the clips gently to the outside with each thumb while gently pulling the SIMM toward you with your index fingers.

4.4.6 Variable Length Enable (VL_EN) (J4)

If the network will be using variable-length message packets, set jumper J5 to ON (Figure 3-7).

Factory default setting: ON.

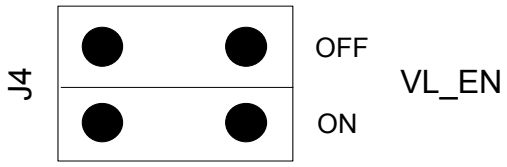


Figure 4-8 Variable Length Messages (J4)

4.4.7 Ground Jumper (J6)

The Chassis/Signal Ground is a user option (Figure 3-8).

Factory default setting: CHASSIS GROUND (⏚).

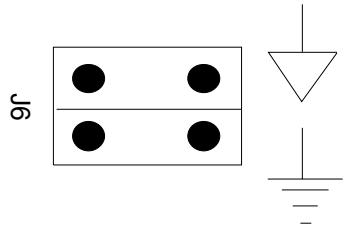


Figure 4-9 Ground (J6)

4.4.8 External Trigger Connections

The SCRAMNet+ board generates two external triggers. Activating the triggers for any shared-memory location will cause an external trigger to be generated when the shared-memory location is accessed (Figure 3-9, Tables 3-5, 3-6).

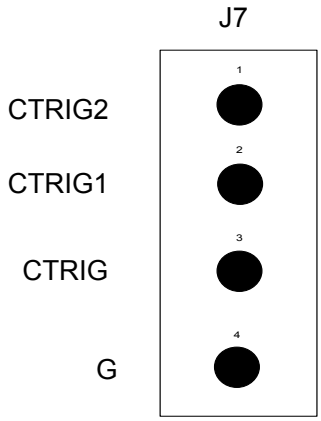


Figure 4-10 External Trigger Connections (J7)

Table 4-4 Trigger Pin Connections (J7)

Pins	Output
1	TRIG2
2	TRIG1
3	TRIG1 or TRIG2
4	GROUND

Table 4-5 External Trigger Actions

Trigger	ACR	Action
1	bit 2	Host READ/WRITE
2	bit 3	Network READ/WRITE

4.5 Install the Board

Once all the switch settings have been made, the board is ready to be installed so the CSRs may be accessed to continue configuration.



CAUTION: Make certain that the power to the host computer is OFF.

The SCRAMNet+ Network node requires one board slot in the backplane. Place the top and bottom edges of the board into the slide-guides of the card cage slots with the component side of the boards facing the same direction as the other existing boards. Slide the board in until it butts against the backplane. Using the levers on the lower end of the faceplate, seat the board in place on the backplane.

The machine is now ready to have power turned ON. Watch the Operator's Console for any booting problems that may occur due to improper settings which correspond to illegal address space or improper seating of the node.

4.6 Select Cabling Options

4.6.1 Cabinet Kit Connection

The SCRAMNet+ board that is configured to work with the cabinet kit has no Media Card of its own. The cabinet kit configuration is discussed in Appendix B.

FIBER-OPTIC CABLE PRECAUTIONS

Fiber-optic cables are made of glass and may break if crushed or bent in a loop with less than a 2-inch radius.

Perform a visual check of the cable ends before inserting into the Media Card connector. If debris is inserted into the transmitter/receiver connector, it may not be possible to clean it out and it could result in damage to the transmitter or receiver lens. Hair, dirt, and dust can interfere with the light signal transmission.

Use an alcohol-base wipe to clean the cable ends.

4.6.2 Fiber-Optic Configuration

The basic SCRAMNet+ Network communication architecture consists of SCRAMNet+ boards tied together by paired sets of fiber-optic cable in a ring configuration. The maximum recommended distance between each node of the network using this configuration is approximately 300 meters. Maximum node separation using long-link fiber is 3,500 meters. The recommended fiber-optic cable is 62.5/125 micron core multi-mode fiber cable with ST connectors.

NOTE: On a freshly powered system, a message from any node on the ring may be necessary to establish the carrier signal.

4.6.3 Coaxial Cable Configuration

SCRAMNet+ coaxial cable is composed of paired, shielded conductors terminated with SMA connectors. Maximum node separation using coax is 30 meters. The recommended coaxial cable is RG-58.

4.6.4 Fiber-optic Cables

The optional paired fiber-optic cables are shipped in a separate carton. The fiber-optic cables are to be attached to the connectors on the SCRAMNet+ board or the Cabinet Kit, as appropriate. Remove the rubber boots on the fiber-optic transmitters and receivers as well as the ones on the fiber-optic cables. These rubber boots should be replaced when cables are not in use or in the event the node must be returned to the factory.

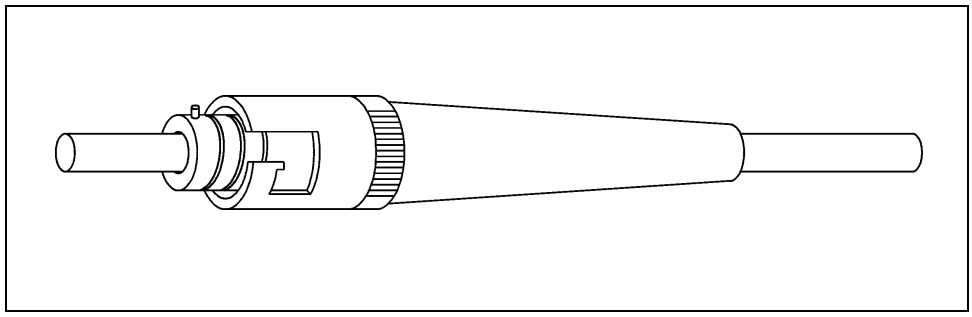


Figure 4-11 Fiber-optic ST Connector

It is important that the ends of the fiber-optic cable be kept clean. If there is an exceptional amount of light-power loss experienced, the cable ends should be inspected for cleanliness. Alcohol-based fiber-optic cleaning pads are available to remove minor contaminants such as dust and dirt.

Figure 4-11 is a representation of a fiber-optic connector.

4.6.5 Fiber-optic Connection

The fiber-optic cable transmitter pairs of the up-stream node are connected to the receiver pair of the down-stream node. Data flows from the transmitter pair of one node to the receiver pair of the next node as described in Figure 4-12.

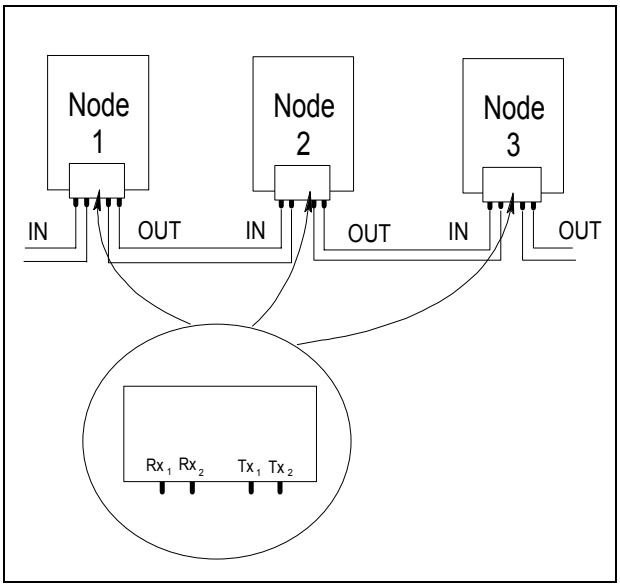


Figure 4-12 Fiber-optic Connections

NOTE: It does not matter if Tx_1 or Tx_2 is connected to the next node's Rx_1 or Rx_2 as long as both Tx cables are connected to both of the next node's Rx connectors

4.7 Install Fiber Optic Bypass Switch (Optional)

Make Fiber Optic Bypass Switch connections as shown in Figure 4-13 and Figure 4-14.

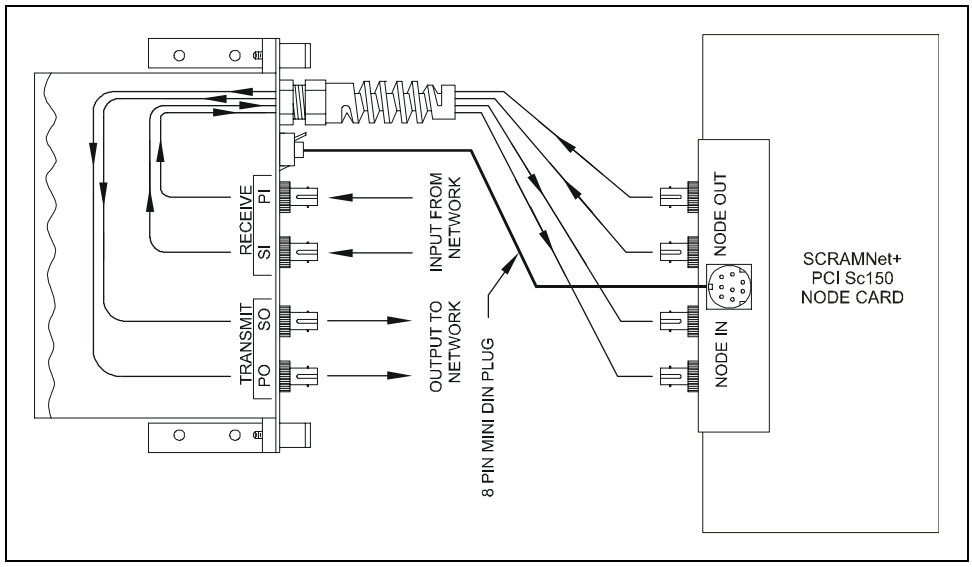


Figure 4-13 Inserted State (Power On)

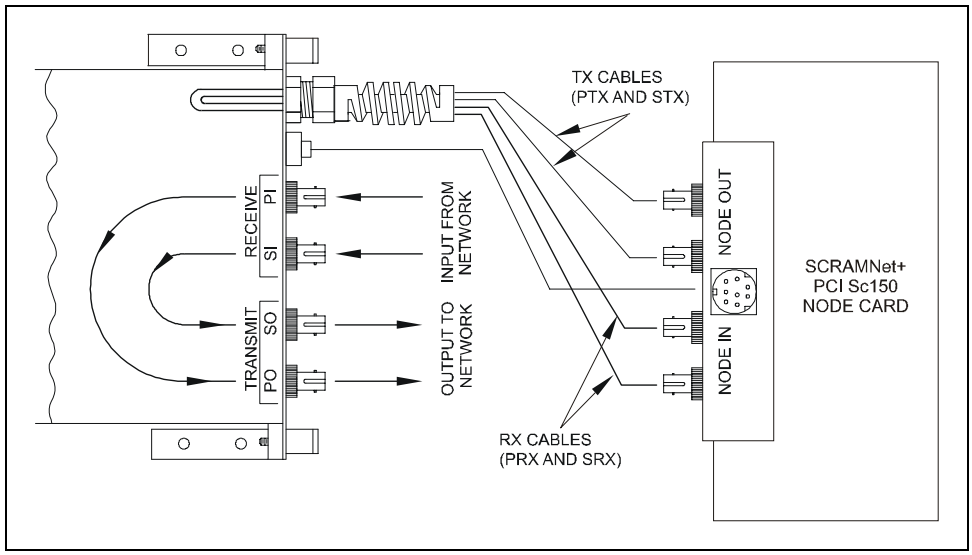


Figure 4-14 Bypass State (Power Off)

4.7.1 Auxiliary Connection

The optional Fiber Optic Bypass switch is used to provide an uninterrupted fiber-optic path when the node is not powered up.

The Auxiliary Connection at the Media Card is used for communication with the Fiber Optic Bypass Switch. The 8-pin modular in-line plug male-pin connection (Figure 4-15) is defined in Table 4-6. (The view in Figure 4-15 is looking into the connector.)

Table 4-6 Auxiliary Connection Pinout

Pins	Name	Definition
1	GND	Logic Ground
2	S_CLK	Serial Clock
3	F_RELAY	Fiber-optic Relay Drive and Sense
4	S_DATA	Serial Data
5	EXT_PWR	+5 Source to External Ground
6	S_DIR	Serial Data Direction
7	TRIGGER	Trigger Output (TRIG1)
8	BAK_PWR	Backup +5 Source from External Device

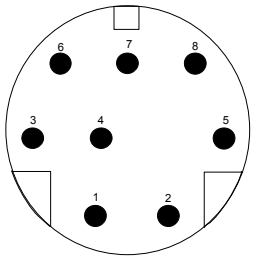


Figure 4-15 Auxiliary Connection

NOTE: Specifications for the Fiber Optic Bypass Switch are contained in Appendix A.

4.8 Internally Configure the Board

Specific items which must be configured through the CSRs include: Node ID, network time-out, memory addressing, shared memory on/off, BURST PLUS protocol on/off, interrupt options, and virtual paging.

INITIALIZATION

The EEPROM is used to store the initial power-up register values. The EEPROM can be programmed either over the host backplane or by most PROM programmers. An EEPROM initialization program is included in the Core Software Package offered by SYSTRAN Corp.

The initialization of the SCRAMNet+ node from a cold boot is determined by the settings of the EEPROM as indicated in Table 4-7.

Table 4-7 EEPROM Table

	0	2	4	6	8	A	C	E
00	0000	0000	C040	XX00	XX00	0010	0000	0000
10	0800	0FF0	0000	0000	0000	0000	0000	0020
...	0000	0000	0000	0000	0000	0000	0000	0000
70	XXXX	YYYY	YYYY	5555	ZZZZ	5555	5555	5555

NOTE: XXXX: 00A1=ASIC1, 00B1=ASIC2; YYYY ignore; ZZZZ = Serial Number

The board comes with power-up default values in each of the registers and default switch settings so the board can be used without making any changes. Defaults are shown in Table 4-8.

Table 4-8 EEPROM Initialization

SCRAMNet+ Registers	
CSR0 - 0	CSR1 - READ Only (Errors)
CSR2 - 0xC040 (BURST Mode)	CSR3 - Node ID (0 - 255)
CSR4 - 0 (READ Only)	CSR5 - 0x0010 READ Only (WRITE Network Time-out to shadow register)
CSR6 - 0 (Data Vector)	CSR7 - 0 (Error Vector)
CSR8 - 0x0800 (Mech Switch Override)$	CSR9 - 0x0FF0 (Error Mask)#
CSR10 - 0 (SM Address LSW)	CSR11 - 0 (SM Address MSW)
CSR12 - 0 (Virtual Page)	CSR13 - 0 (GP Counter)
VME Registers	
CSR14 - 0 (Reserved)	CSR15 - 0x20 (Interrupt Level 5)
CSR16 - 0 (HIPRO READ)	

$ VME HOLDOFF is enabled
# Normal Error Masking enabled

These values are assigned to the Control/Status Registers on power-up. The values may be changed at any time using the appropriate software to access the CSRs. When the system is powered down and powered up again, the CSRs will be reinitialized to these EEPROM values.

4.8.1 Node Identification

All nodes in the ring must have a unique Node ID. To set the Node ID, WRITE a unique value 8-bit number between 0 and 255 to CSR3[15:8].

4.8.2 Network Time-out

The recommended value for the Network Time-out is:

 # NODES IN RING + (TOTAL LENGTH OF CABLE IN METERS÷50) + 1

To set the network time-out value, WRITE the result as a 16-bit hexadecimal number to CSR5. This register has a "shadow register" which holds the network time-out value. Each increment in the "shadow register" is worth approximately 240 ns. The time-out will be 240 ns multiplied by the value written.

The time-out register is WRITE ONLY. If a READ is performed, it will result in a READ to the Interrupt FIFO.



CAUTION: Ensure a non-zero value is written to CSR5. A value of '0' will keep host-generated data from leaving the Transmit FIFO

4.8.3 Memory

MEMORY ADDRESSING

The memory base address is software configurable anywhere through the A24 and A32 address space. If the shared-memory and CSR addresses should overlap, the CSR s will be decoded making the overlapped memory inaccessible.

The Least Significant Word (LSW) of shared-memory address is in CSR10[15:12]. The Most Significant Word (MSW) of shared-memory address is contained in CSR11[15:0].

SHARED MEMORY

SMA_ENABLE CSR10[0] must be set ON to enable the shared-memory address decoder.

4.9 Byte Swapping

Some computer systems use different methods of byte ordering. Some have the high bits arranged from right to left (Little Endian) and other have the high bits going from left to right (Big Endian). Motorola is an example of a Big-Endian system. Intel is an example of a Little-Endian system.

Table 4-9 is a simplified summary for 8-bit, 16-bit, and 32-bit byte ordering for big endian and little endian.

Table 4-9 Big Endian - Little Endian Comparisons

Size	Big Endian	Little Endian
byte (8-bit)	12 34 56 78	78 56 34 12
shortword (16- bit)	1234 5678	5678 1234
longword (32-bit)	12345678	12345678

The SCRAMNet-LX and SCRAMNet+ product line has adopted the big-endian ordering philosophy as the default for data passing. SCRAMNet-LX and SCRAMNet+ do not have a built-in byte-ordering conversion function.

4.10 Troubleshooting

On UNIX-like systems, the driver will output a message on boot up similar to the following:

```
SCRAMNet+ installed and on line
```

If this message does not appear then one or more of the following is not correct:

- Physical address for the CSRs switch selection on the board.
- VME address bus for the CSRs (16- or 24-bit)
- The address for the CSRs in the system files is not correct.
- The driver was not installed into the new kernel properly.

4.10.1 Other H/W Considerations

- The SCRAMNet+ Network is not installed into a standard VME slot. Some vendors mix VME and their own private bus in what appears to be the same backplane. This check should be performed before installing the SCRAMNet+ Network card into any slots.
- The Bus Resolution switch (S8) is set incorrectly. See Table 4-3 on page 4-6.
- The memory bus address and/or bus size is set incorrectly.
- The CSR bus address and/or bus size is set incorrectly.
- The interrupt request level is set too high (7).
- The memory is installed at an improper boundary for the size of memory. For example, a 2 MB board can not be installed at hex address 10500000. However, 10400000 or 10600000 are on 2 MB boundaries.

4.10.2 Customer Support

If the system does not boot correctly, reseat the board and double-check cable connections. If problems persist, call Systran Customer Support at (937) 252-5601 for assistance.

Please be prepared to supply the following information:

Host machine: _____
Operating System
Name: _____
Version: _____
Bus Interface: _____

This page intentionally left blank

5. OPERATION

5.1 Overview

The SCRAMNet+ Network is a shared-memory system. Every computer on the network has a constantly updated local copy of all global data that is passed to all the network computers. The network protocol is implemented in the SCRAMNet+ hardware and therefore no software overhead is required to retrieve this information from the network.

The protocol is transparent to the computer. This frees computer processor time for application algorithm execution and other real-time tasks. Since any computer on the network has access to data in the shared memory, any computer can read or modify the shared data and thereby communicate with the other computers on the network.

Very little special software is required for normal operation because of the SCRAMNet+ shared-memory configuration. Typically, SCRAMNet+ memory is installed and linked to a host global common block through the host operating system. Once the link is complete, any program can reference SCRAMNet+ memory as a standard common-block variable reference.

For interrupt driven applications, an interrupt service routine (ISR) is required to handle the interrupts triggered by the SCRAMNet+ node. An example of a generic ISR is included at the end of this section, Figure 5-11, page 5-29.

5.2 Shared Memory

Global variables are mapped directly onto the replicated shared memory. The application program typically contains a list of variables or arrays which are stored in a contiguous space and which are to be shared across processors. The analogy of a FORTRAN COMMON BLOCK is most fitting. For the purpose of identification, these variables are referred to as SCRAMNet+ variables.

The application program usually requires a short section of instructions to initialize the SCRAMNet+ hardware and to link the SCRAMNet+ memory to the SCRAMNet+ variable list. The shared memory cannot be used as instruction space.

5.2.1 Virtual Paging

CSR12 is the virtual-paging register. Set CSR12[0] to ‘1’ to enable virtual paging.

All SCRAMNet+ nodes use the same 8 MB network shared-memory map. Virtual paging allows a node with less than 8 MB shared memory to move their memory window throughout the SCRAMNet+ physical 8 MB shared-memory map.

If a node has 4 MB of shared memory, it can be paged into the upper 4 MB or the lower 4 MB of the shared-memory map. If it is paged into the lower 4 MB, it would operate the same as if Virtual Paging were disabled. The network address would be the same as the shared-memory address.

NOTE: Virtual paging does not affect host access to shared memory. Virtual Paging only changes the network address. The HOST SPECIFIC logic always sees the base address of SCRAMNet+ shared memory as zero.

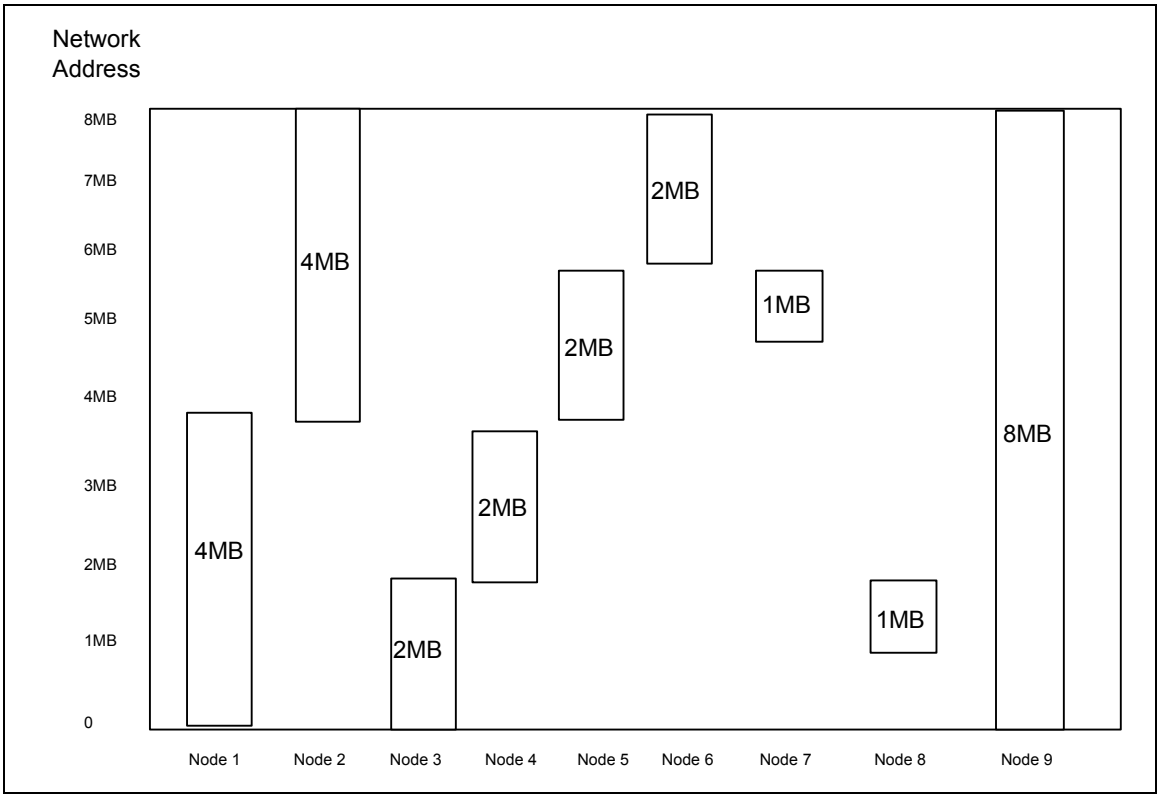


Figure 5-1 Memory Sharing With Virtual Paging

If the 4 MB block were paged into the upper 4 MB, a host-specific WRITE to a shared-memory address of 2 MB would result in a network address of 6 MB. This translation is bi-directional. An incoming network message with a network address of 6 MB will be written to shared memory at 2 MB. Any WRITE accesses to the lower 4 MB will be ignored since there is no memory addressed there.

To produce a network address, the host WRITE adds the relative SCRAMNet+ address and virtual-page offset.

relative address + virtual-page offset = network address
For example: 12340 + 400000 = 412340

This network address is then transmitted to all of the SCRAMNet+ nodes and is written to that address. In nodes where the address does not exist in SCRAMNet+ memory, the WRITE is ignored.

The example in Figure 4-1 shows the memory-sharing relationships between various nodes in the virtual-paging mode. Node 1 shares data with nodes 3, 4, 8, and 9. Node 2 shares data with nodes 5, 6, 7, and 9. Node 8 shares data with nodes 1,3, and 9. Node 7 shares data with nodes 2, 5, and 9. Node 9 shares data with all nodes.

5.2.2 Memory Considerations

When using SCRAMNet+ shared-memory, consider the following:

PROGRAM AND DATA LIMITATIONS

Limitations on application program size and data variable size for a host computer system also apply to applications that use SCRAMNet+ memory because it becomes part of the host system.

DATA CACHING

The ability for a computer to write a copy of data to a local fast memory for quicker access later must be turned off during a SCRAMNet+ memory read. Since other nodes may be changing the data, it is critical that the processor read the data directly from SCRAMNet+ memory. This is processor dependent and does not always apply.

MEMORY MAPPING

SCRAMNet+ memory is mapped by all operating systems in constant-length blocks called memory pages.



NOTE: To ensure that a compiler or operating system does not try to use unused portions of SCRAMNet+ memory to store other program segments, the SCRAMNet+ memory common blocks should be declared to be sized to an integer multiple of the processor memory page size. If this is not done, most compilers will try to optimize memory usage by filling out the SCRAMNet+ memory pages with other data. This can cause random results when this local data is transmitted around the network.

5.2.3 Control/Status Registers

The SCRAMNet+ boards are controlled through CSRs for node status, setting interrupt vectors, setting interrupt locations, receiving interrupt addresses, mode control and other functions. These registers may be accessed by linking to the I/O page and reading from or writing to the registers as if they were memory. The method used to access the registers depends on the particular computer and operating system being used.

These registers are set only during the SCRAMNet+ Network initialization. Once the control portion of the CSR is set up for the desired mode operation, the node functions as transparent shared memory, and references to the CSRs are not required. However, the status portions of the registers will need to be accessed for interrupt servicing and for checking for error conditions. Appendix B, CSR DESCRIPTIONS, discusses the definition and use of each bit in the CSRs. Appendix C contains a list of the CSRs and a brief identification of each bit.

5.3 Initialization

The initialization of the SCRAMNet+ node from a cold boot is determined by the settings of the EEPROM (see Chapter 4, INSTALLATION).

No fiber-optic cable connections are required to READ and WRITE to the local host's SCRAMNet+ memory. The control registers CSR0 and CSR2 should both be zero at this point, and SCRAMNet+ memory is available for access. The memory address will remain at '0' and be disabled until programmed with the EEPROM Initialization Program.



NOTE: All SCRAMNet+ nodes in the fiber-optic network ring must be powered on unless Fiber Optic Bypass Switches or Quad Switches are installed.

5.4 Basic Send/Receive Configuration

The following conditions must exist in order to have basic send and receive capability on the network:

- Set CSR0 to '0xF000' to insert the node and initiate the reset of the FIFOs.
- Set CSR0 to '0x8003' to insert the node, toggle the reset of the FIFOs and enable network activity.
- Set CSR2 to '0xC040' to use BURST mode and disable the fiber-optic loopback mode.
- READ CSR1 to read-out any latched error conditions.
- READ CSR1 again to check for any existing error conditions.
- Check for carrier-detect fail (this means there are fiber-optic cabling problems from the transmitter of the node downstream).
- WRITE a value to memory from at least one node. This enables all powered node transmitters and checks for fiber-optic ring integrity.
- READ CSR1 to check for any error conditions.

5.5 Network Ring

Data is passed from one node to the next by fiber-optic or coaxial cable. Given a three-node network configuration with nodes A, B and C, the following connections would be made:

- The transmitter pair from node A is connected by fiber-optic cable to the receiver pair of the next node B.
- The transmitter pair from node B is connected by fiber-optic cable to the receiver pair of node C.
- The transmitter pair of node C is then connected to the receiver pair of node A, thus completing a fiber-optic network ring.

5.5.1 Message Contents

The smallest SCRAMNet+ Network message packet consists of 82 bits. This basic message format contains five fields: Source ID, Age, Control, Data Address, and Data Value. The message can be described as follows:

Table 5-1 SCRAMNet+ Message Contents

START	ID	AGE	CONTROL	DATA ADDRESS	DATA VALUE
1	8+P	8+P	1 1 1 RES INT RTY	5+P 8+P 8+P	8+P 8+P 8+P 8+P

For every 8 bits of data in the message there is a parity bit attached.

SOURCE ID

This 8-bit field contains the node ID of the originating node. Value ranges from 0 to 255, so there can be 256 nodes on the network ring.

AGE

This 8-bit field increments by one as a message passes through each network node. If the age ever exceeds 256 (the maximum number of nodes on the network), the message is removed from the network.

CONTROL BITS

RES - Reserved.
INT - When this bit is set it signals an Interrupt Message.
RTY - Retry message used only in error correction mode (PLATINUM.)

DATA ADDRESS

This 21-bit (A[22:2]) field contains the relative SCRAMNet+ memory address. Bits A0 and A1 are always zero for a longword boundary.

DATA VALUE

This 32-bit field contains the data value in SCRAMNet+ memory that is currently being updated around the ring. When the PLUS mode is enabled, data size may vary up to 256 bytes or 1024 bytes depending on the option selected.

5.5.2 Protocol

BURST MODE

BURST Mode is the normal protocol for SCRAMNet+. The BURST mode is enabled by setting CSR2[12] OFF, and CSR2[15:14] ON. The BURST Mode protocol allows each node to continuously transmit messages onto the network ring. This mode uses a 4-byte fixed-length message packet for data transfer.

PLATINUM MODE

The PLATINUM mode is BURST mode with error correction enabled. PLATINUM mode is enabled by setting CSR2[12] OFF; CSR2[14] ON; and CSR2[15] OFF See the Protocol Mode Definition table, Appendix B, page B-8.

PLUS MODES

The PLUS mode protocol is available as an option to the SCRAMNet+ BURST and PLATINUM mode network protocols. Selecting the 1024 byte maximum data length will result in the maximum bandwidth. However, it will also lengthen ring time, error, correction time, and node latency. Set up the PLUS mode protocol as follows:

- Set CSR2[12] to 1 to enable PLUS mode.
- Set CSR2[11] to define the maximum data message size. CSR2[11] only has an effect when CSR2[12] is ON. CSR2[11] has the following definition:
  0 = 256 byte maximum data length
  1 = 1024 byte maximum data length
- Set up the WRITEs to SCRAMNet+ shared memory to be in sequential addresses. Data must be written to the SCRAMNet+ node with sequentially incrementing 32-bit addresses to take advantage of the PLUS mode protocol enhancement. It is not necessary to use the full 256 or 1024 byte data length—they are maximum values.

The PLUS mode allows variable-length message packets in which sequentially addressed data in the Transmit FIFO is transferred as a block in a single packet. Both BURST modes are open loop, non-error-corrected modes of operation.

The node appends 4-byte data values with sequential addresses until the maximum of 256 or 1024 bytes is reached, a non-sequential address is detected, the Transmit FIFO is empty, or a transmit interrupt event is detected. In both BURST and PLATINUM modes, the node is permitted to have multiple packets on the ring simultaneously.

The transmission of a PLUS mode message is an automatic function, and for the most part, cannot be controlled. If the appropriate PLUS mode bits are set in the control registers, then the following algorithm applies:

1. If Transmit FIFO is empty, end transmission.
2. If the address field is not equal to the address of the previous transmission + 4, end transmission.
3. If length limit overflow for PLUS mode operation occurs, end transmission.
4. ELSE transmit the four data bytes and when done GOTO step 1.

To maintain a PLUS mode transmission, step 1 requires that new data is written to the SCRAMNet+ board at a rate greater than or equal to 16.7 MB/sec; this is a 32-bit WRITE every 240 ns. Any delay in the host data WRITE will result in failure of step 1, and a premature end to the PLUS mode transmission.

While this method results in the reliable generation of a PLUS mode transmission, it increases the node latency. The SCRAMNet+ device automatically increases PLUS mode throughput (when blocking is not used) when needed due to high-throughput host, very busy network, etc.

ERROR CORRECTION

Error correction is the automatic retransmission of a SCRAMNet Network message when the original message is received in error by the originating node. The message will be retransmitted indefinitely until it is received correctly. During transmit retry, the same message is being sent. This prevents any new messages from being transmitted by this node. The Transmit FIFO will hold these new messages until the retry message is received correctly.

If the original message is received by the originating node with some type of bit error, then this results in the transmit retry bit in CSR1 being set. If the original message is not received by the originating node in the time-out period specified in CSR5, then this results in the transmit retry time-out bit in CSR1 being set. The time-out period is based on the number of nodes in the network ring and the total length of cable used.

5.5.3 Performance

NODE LATENCY

Node latency is an important factor in networked application in real-time systems design. Data transfer around the network, while fast, does have a measurable delay.

Node latency can be defined as the time delay at a node before a foreign message can be retransmitted. This delay is a minimum of 247 ns; the time to transmit one byte. The maximum node latency depends on the maximum message size and could be from 800 ns to 61.8 µs, depending on the message length selection. To approximate the total maximum delay on the network, multiply the maximum node latency by the number of

nodes in the system, and add a propagation delay of 5 ns/meter multiplied by the total message path of the ring in meters.

DATA TRANSFER

While the SCRAMNet+ Network appears as a shared-memory system, it is still a data network. The SCRAMNet+ Network includes a series of FIFO buffers to collect data changes until they are transmitted to the other nodes. The Transmit FIFO and the Interrupt FIFO are both 1024 messages in length. These numbers may become significant when performing data transfers of large blocks of data in a short period of time.

VME HOLDOFF

If the Transmit FIFO becomes full, subsequent READ or WRITE cycles to SCRAMNet+ memory will be extended until the Transmit FIFO is no longer full (see paragraph 5.11.5 for more information).

SHARED-MEMORY WRITE

SCRAMNet+ shared-memory is based upon a 32-bit word. If an 8- or 16-bit WRITE occurs from the host system, then the 32-bit word that contains that 8- or 16-bit WRITE is sent on the network. Therefore, it is important that other nodes do not simultaneously modify other 8- or 16-bit segments within that 32-bit word.

5.5.4 Throughput

A maximum throughput of 6.5 MB/sec could be achieved if only one node were transmitting data, assuming the host CPU could offer the data at that rate. When more than one node is transmitting in BURST mode, then the effective output per node is 6.5 MB/sec divided by the number of transmitting nodes. In BURST and BURST PLUS modes, the node never retransmits its own messages.

In the BURST PLUS mode, a 256-byte packet provides 16.2 MB/s of data throughput. A 1024-byte packet provides 16.7 MB/s maximum data throughput.

When multiple nodes are transmitting in the BURST mode, the network data passing through the other nodes can affect that node's output performance. If a node's receiver is so busy that the Transceiver FIFO is never empty, and the node has already sent a message, then the node will have to wait before it can send another message of its own until either one of its messages comes back or the timer runs out. When the node's own message is received, it is not placed in the Transceiver FIFO thereby creating an opportunity for the node to send a message from the Transmit FIFO.

In PLATINUM and PLATINUM PLUS modes, error detection is enabled. This will affect node latency in that some messages must be retransmitted.

NETWORK TIME-OUT

Reset the transmit time-out according to the mode of operation selected by writing a 16-bit, non-zero value to CSR5 as described in paragraph 3.8.2.

5.6 Auxiliary Control RAM

The ACR is an 8-bit register. However, only ACR[4:0] are implemented. ACR[7:5] are not defined.

When ACR Enable CSR0[4] is set, shared memory is not accessible by the host, and the ACR byte is viewed as the least significant byte of every shared-memory four-byte

address. The ACR byte value controls the external trigger and/or interrupt action(s) taken whenever a WRITE occurs to any byte of the shared-memory 4-byte word. Table 5-2 describes the ACR functions.

If these ACR actions are disabled, then no action will be taken when an interrupt condition exists unless override bits CSR0[6] or CSR8[10] are set.

The external trigger and/or interrupt action and/or HIPRO mode for a particular shared-memory location is defined by setting these bits. Once the ACR has been defined, set ACR Enable bit CSR0[4] back to zero so that shared memory can again be accessed. The ACR actions are still in effect, but the ACR bytes can no longer be accessed while the ACR Enable bit is zero.

Table 5-2 ACR Functions

Bit	Function
0	Receive Interrupt Enable (RIE)
1	Transmit Interrupt Enable (TIE)
2	External Trigger 1
3	External Trigger 2
4	HIPRO Location Enable
7-5	Reserved

In order for the ACR values to take effect for interrupt action, the following SCRAMNet+ CSR actions should be considered for the type of interrupt operation desired:

- Host Interrupt Enable CSR0[3] to receive network interrupts
- Network Interrupt Enable CSR0[8] to transmit network interrupts
- Interrupt on Memory Mask Match Enable CSR0[5] for interrupts from memory WRITEs

Receive and/or Transmit CSR0[1:0] must be enabled in order for the node to receive and/or transmit network data. There are other combinations of CSR settings to achieve varied interrupt results. Section 5 describes the SCRAMNet+ CSRs in detail.

In order for the external triggers 1 and 2 to take place, ACR[2] and ACR[3], respectively, must be set. In order for the HIPRO mode to become active, ACR[4] must be set for those selected memory addresses where this is to occur. Additionally, CSR2[13] must be set to enable the HIPRO mode. All five of the defined bits of the ACR can be used in any combination to achieve varied results for any shared-memory location.

5.7 Interrupt Controls

SCRAMNet+ allows a processor to receive interrupts from and/or transmit interrupts to any other processors on the network, including the originating processor. Table 5-3 indicates the various sources for interrupt control.

5.7.1 Interrupt Options

Table 5-3 Interrupt Controls

Condition	Register	Description
Host Interrupt Enable	CSR0[3]	Must be set in order to receive any interrupts from the network.
Receive Interrupt Enable (RIE)	ACR[0]	Generates an interrupt to the host from network data received at the associated shared-memory location.
Transmit Interrupt Enable (TIE)	ACR[1]	Generates an interrupt message to the network for a host WRITE to the associated shared-memory location.
Interrupt on Memory Mask Match Enable	CSR0[5]	Permits a shared memory interrupt. Must be set in order to receive any interrupts from the network.
Override RIE	CSR0[6]	Generates an interrupt to the host regardless of the ACR RIE setting upon receipt of any network interrupt message.
Enable Interrupt on Error	CSR0[7]	Generates an interrupt request as specified in the CSR9 Mask register as the corresponding bit in CSR1 is set.
Network Interrupt Enable	CSR0[8]	Permits transmission of interrupt data to the network.
Override TIE	CSR0[9]	Transmits interrupt message to the network regardless of the ACR TIE setting.
Reset Interrupt FIFO	CSR0[13]	Toggle from ‘0’ to ‘1’ to ‘0’ to reset Interrupt FIFO.
Interrupts Armed	CSR1[14]	During the interrupt operation indicates conditions to receive interrupt are active. If ‘0’, no interrupts will be received by the host. Any WRITE to CSR1 will reset to ‘1’.
Enable Interrupt on Own Slot	CSR2[10]	In conjunction with CSR2[9] enables host self-interrupt.
LSP of Interrupt Address	CSR4[15:0]	Interrupt Address A[15:0].
MSP of Interrupt Address	CSR5[6:0]	Interrupt Address A[22:16]. Works in conjunction with CSR4[15:0].
Interrupt FIFO Not Empty	CSR5[15]	When ‘0’, Interrupt FIFO is empty. If ‘1’, CSR5 and CSR4 contain legitimate interrupt address.
Receive Interrupt Override	CSR8[10]	When set, all incoming network messages are treated as interrupt messages.
Interrupt on Error Mask	CSR9[15:0]	Interrupts for specific error/status conditions .
VME Interrupt Priority Level (IRQ)	CSR15[7:1]	7-bit, host-specific register that holds the VME Interrupt Levels

SEND/RECEIVE WITH INTERRUPTS

- Set CSR0 to '0x0010' to enable the Auxiliary Control RAM (ACR).
- Clear the SCRAMNet+ ACR by writing zeros to the entire address range.
- Set the SCRAMNet+ ACR memory locations designated to receive and/or transmit interrupts.
- Reset CSR0 to disable the ACR.
- Set CSR0 to '0xF000' to insert the node and initiate the reset of the FIFOs.
- Set CSR0 to '0x8003' to insert the node, toggle the reset of the FIFOs and enable network activity.
- Set CSR2 to '0xC040' to use BURST mode and disable the Fiber-optic Loopback mode.
- READ CSR1 to read-out any latched error conditions.
- READ CSR1 again to check for any existing error conditions.
- Check for carrier-detect fail (this means there are fiber-optic cabling problems from the transmitter of the node downstream).
- WRITE a non-interrupt value to memory from at least one node. This will enable all powered node transmitters and check for fiber-optic ring integrity.
- READ CSR1 to check for any error conditions.
- Set CSR6 to the proper interrupt vector number.
- Install the interrupt service routine.
- Set CSR0 to '0x812B' to enable receive and transmit interrupts.
- Set CSR0 to '0x81AB' to enable Interrupt On Errors.

5.8 Interrupt Conditions

Interrupts are generated under two different conditions:

- A SCRAMNet+ network data WRITE to shared memory
- A SCRAMNet+ network error/status detected on the local node

5.8.1 Network Data WRITE

As indicated in Figure 5-2, Transmit Enable CSR0[1] must be set before any message can be sent. Only those nodes which have Transmit Interrupt Enable ACR[1] set for selected addresses send an interrupt bit out with the data packet on the network. Only those nodes which have Receive Interrupt Enable ACR[0] set for that address will generate an interrupt signal to their host processor.

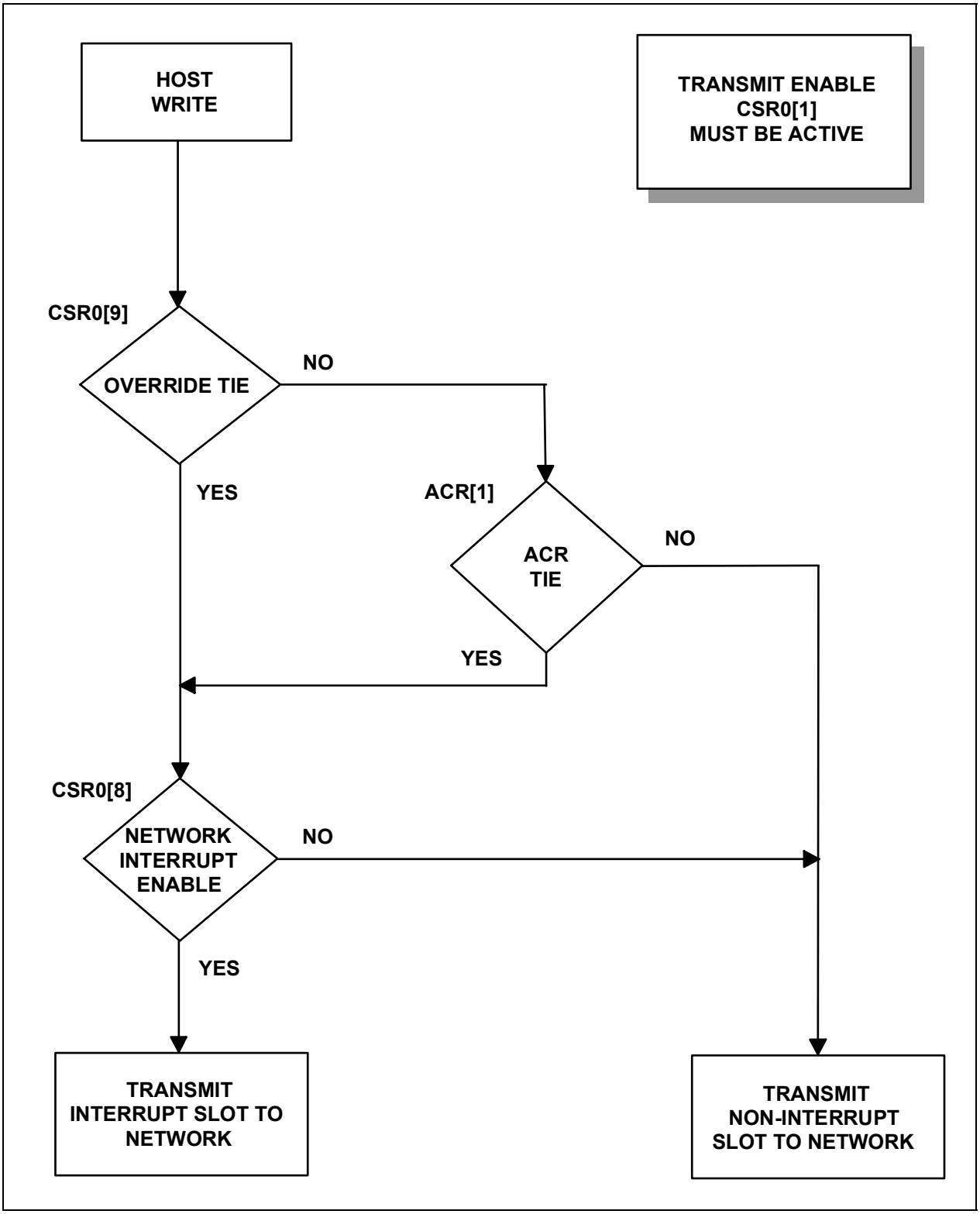


Figure 5-2 Transmit Interrupt Logic

The host issues a WRITE to SCRAMNet+ shared memory. If Override TIE CSR0[9] or ACR TIE ACR[1] is set and Network Interrupt Enable CSR0[8] is set, then the interrupt message is transmitted (INT = 1). Otherwise, the message is transmitted without the interrupt bit set (INT = 0). (See Table 5-1, page 5-4)

Network data WRITE interrupts can be accomplished by two methods:

- **Forced**. Any data WRITEs to any shared memory from the network will generate an interrupt.
- **Masked or Selected.** Data writes to selected shared-memory locations from the network. Under either of these two methods, an interrupt can be generated and received by the same host processor if desired. This condition is called Self-Interrupt.

FORCED INTERRUPT

The forced interrupt method works the same as the selected interrupt method with the exception of choice of interrupt locations. All shared-memory locations are automatically set up to receive and/or transmit interrupts depending upon the override bits set in CSR0 or CSR8.

MASKED OR SELECTED INTERRUPT

The masked or selected interrupt method requires choosing SCRAMNet+ shared-memory locations on each node to receive and/or transmit interrupts. These shared-memory locations may also be used to generate signals to external triggers. The procedure for selecting shared-memory locations for interrupts and/or external triggers is explained in paragraph 5.6: Auxiliary Control RAM.

CSR5 contains the Interrupt FIFO Not Empty bit CSR5[15].

SELF-INTERRUPT

Set CSR2[10:9] to enable self-interrupt. This allows the message with the interrupt bit set to be processed as an incoming network interrupt. CSR2[9] enables the node's own message to be received as a network message. CSR2[10] allows the interrupt bit to generate an interrupt if it is set.

Receive Interrupt logic is described in Figure 5-3. If a native message is received and Write Own Slot CSR2[9] is enabled, and Enable Interrupt on Receipt in own Slot CSR2[10] is set, the logic then checks for Receive Interrupt Enable. If Override RIE CSR0[6] is set or ACR RIE ACR[0] is set, and if Interrupt Mask Match Enable is set, the address is placed on the Interrupt FIFO.



NOTE: Interrupt data is not filtered when the data filter is enabled

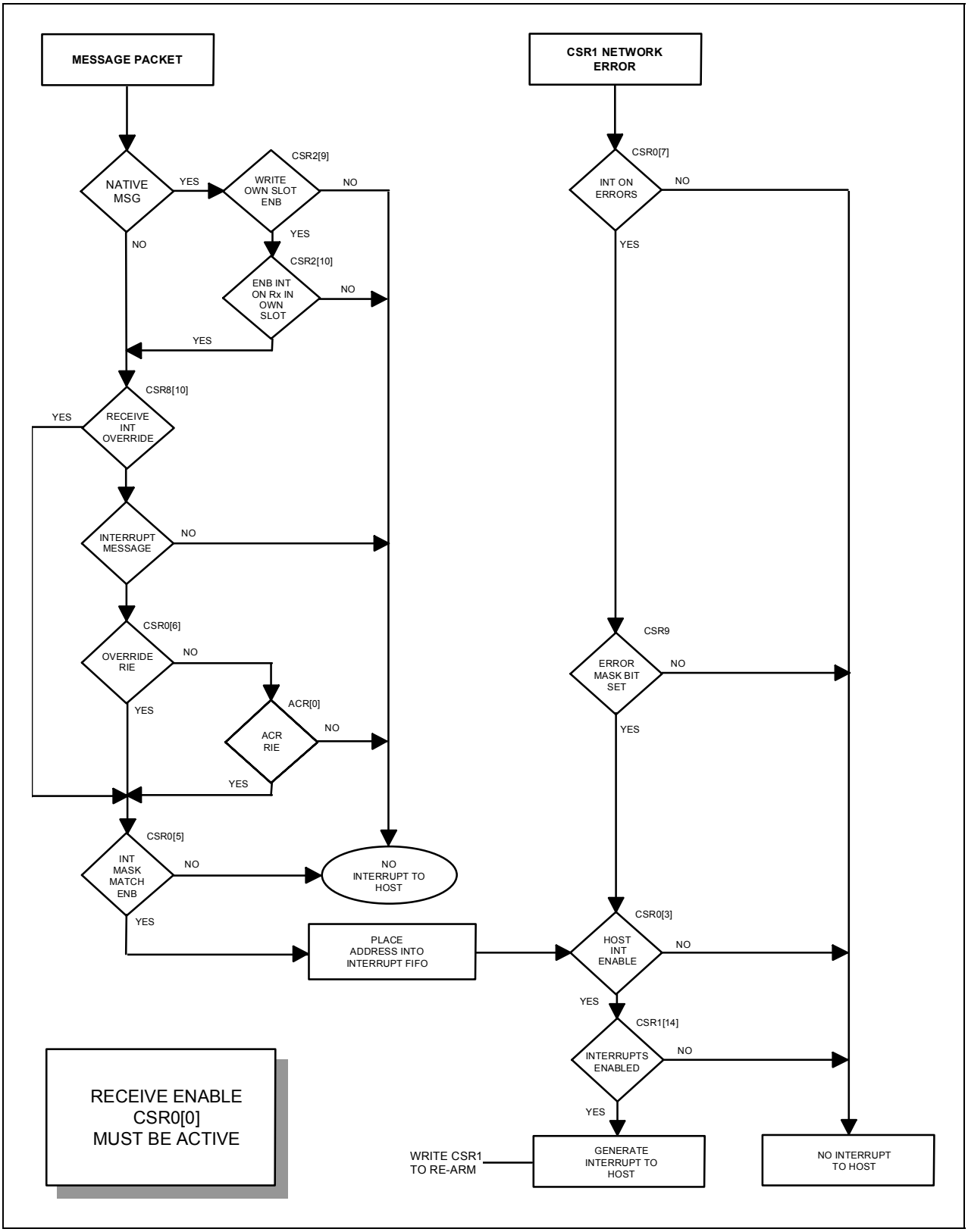


Figure 5-3 Receive Interrupt Logic

5.8.2 Network Error

The second interrupt condition is designed to intercept network errors. CSR1 contains the following error conditions that may be masked by CSR9:

Table 5-4 Interrupt Error/Status Conditions

Bit	Interrupt
0	Transmit FIFO Full
1	Transmit FIFO Not Empty
2	Transmit FIFO ⅞ Full
3	(Not masked for errors)
4	Interrupt FIFO Full
5	Protocol Violation
6	Carrier Detect Failure
7	Bad Message
8	Receiver Overflow
9	Transmit Retry
10	Transmit Retry Time-out
11	Redundant Rx/Tx Fault
12	General Purpose Counter/Timer
13	(Not masked for errors)
14	(Not masked for errors)
15	Fiber Optic Bypass Not Connected

Each of these conditions is identified by the corresponding bit being set (value 1) in CSR1. If any of the preceding conditions are set and Interrupt On Memory Mask Match Enable CSR0[5] is set, then an interrupt will be generated to the host computer. Additional information about each error condition is contained in Appendix B, Table B-2: CSR1.

If a Network Error is received (Figure 5-3), and if Interrupt on Error CSR0[7] and Host Interrupt Enable CSR0[3] are set, and Interrupts are Enabled CSR1[14], then the message generates an interrupt to the host. If additional network data interrupts occur before the processor is able to service the interrupt, those shared-memory locations are updated and the addresses are added to the Interrupt FIFO queue. However, no additional interrupt signals are sent to the host until interrupts are armed by writing to CSR1.

There must also be an interrupt vector to the Interrupt Service Routine. The data vector is stored in CSR6, and the error vector is in CSR7. CSR6 is associated with a memory update and CSR7 is used to identify an error interrupt. Details are included in the *Programmer's Reference Guide* for the host computer interface.

5.8.3 Interrupt Handling

The Interrupt FIFO is accessed via CSR4 and CSR5. CSR5 contains the most significant seven bits of the 23-bit SCRAMNet+ interrupt address and CSR4 contains the remaining 16 bits of the interrupt address. (The 23-bit address allows for future expansion of memory). CSR5 also contains Interrupt FIFO Not Empty (bit 15).



NOTE: The SCRAMNet+ Network is a longword (32-bit)-oriented shared memory. External Triggers and Interrupts will occur when any of the four bytes associated with a long word are accessed. The Interrupt FIFO contains the longword address. If each of the four bytes of an interrupt location are written into as byte accesses, then four interrupts to the same longword address will be generated. Likewise, if each word of an interrupt location is written into as 16-bit shortwords, then two interrupts to the same longword address will be generated.

The two values of CSR5 and CSR4 make up the interrupt address. When an interrupt is received, the ISR should READ CSR5 first in order to check the Interrupt FIFO Not Empty bit. If this bit is set (value is '1'), then READ CSR4. If this bit is CLEAR (value is 0) then the Interrupt FIFO is empty. Therefore, the interrupt was due to an error, assuming that Enable Interrupt On Error is set.

Every READ from CSR5 and CSR4 will contain the SCRAMNet+ memory address of the data received from the network interrupt. Every READ of CSR5 and CSR4 will automatically increment the FIFO pointer to the next interrupt address for both registers. CSR4 should be read only if Interrupt FIFO Not Empty CSR5[15] is set. Continue to READ CSR5 and CSR4 until the Interrupt FIFO Not Empty bit is zero. Writing any value to CSR1 will re-enable interrupts. See Page 5-29 for an example of a standard ISR algorithm for handling interrupts from the SCRAMNet+ boards.



WARNING: If HIPRO is enabled, an interrupt may affect the sequence of addresses on a READ/WRITE if SCRAMNet+ is manipulated in the ISR.

If an interrupt occurs before the interrupts have been armed, the interrupt will be placed in the Interrupt FIFO and it will occur when the interrupts are armed (CSR 1).

5.9 External Triggers

Two external triggers are provided by the SCRAMNet+ Network. The external triggers will occur only if the ACR has been configured to enable them. Triggers 1 and 2 are generated by SCRAMNet+ shared-memory access. Triggers generate a 26.64 ns TTL level compatible, non-terminated, output.

- Trigger 1 - Host READ/WRITE (ACR[2] enables)
- Trigger 2 - Network WRITE (ACR[3] enables)

Trigger 1 will be generated for any host access to SCRAMNet+ memory.
Trigger 2 will be generated by a network WRITE to the SCRAMNet+ memory.

The trigger output signals are available through the external trigger connection pins, if installed on the board. Also, pin 7 of the Auxiliary Connector is connected to TRIG1.

The triggers can be used to measure time intervals or to start or stop an external event.

EXAMPLE 1 – MEASURE TRANSACTION TIME

Select a shared-memory address in node A and enable trigger 1 by setting ACR[2]. Select the same memory address in node B and enable trigger 2 by setting ACR[3]. Connect a wire from TRIG1 to an oscilloscope. Connect a wire from TRIG2 to the oscilloscope. Initiate a HOST WRITE to the node A memory address. When the corresponding NETWORK WRITE occurs, the time difference can be measured.

EXAMPLE 2 – MEASURE RING TIME

Enable WRITE-ME-LAST mode on node A. Select a shared memory address and enable trigger 1 and trigger 2 for that address. Connect a wire from TRIG (pin 3 - trigger 1 OR 2) to an oscilloscope. Initiate a HOST WRITE to that memory address. The time difference displayed on the oscilloscope will approximate the ring time—the time it takes from the HOST WRITE on node A to the NETWORK WRITE on node A.

EXAMPLE 3 – SET OFF AN ALARM

Enable trigger 1 for the shared memory address of a critical datum by setting ACR[2]. Connect a wire from TRIG1 to an alarm light. When the HOST WRITE to that memory address occurs, the light will come on.

5.10 General Purpose Counter/Timer

This 16-bit counter/timer can be programmed by changing CSR9[13] and CSR9[14] to select the desired mode as described in Table 5-5. CSR8[9] can be set to override the counter/timer mode settings and allow the counter/timer to run free at 26.66 ns (37.5 MHz). CSR9[12] can be set to generate an interrupt upon overflow of the counter/timer. The output from the event counter/timer is stored in CSR13. See Appendix B, pages B-6, B-12, and B-14 for more information.

5.10.1 Available Modes

The General Purpose Counter/Timer register (CSR13) can be used as a counter or a timer. The mode is selected via a combination of registers and bits that are explained on page 5-13. Table 5-5 describes the counter/timer modes available:

Table 5-5 General Purpose Counter/Timer Modes

Mode	Description
Count Errors:	Each error detected in CSR1 will increment the counter by 1.
Count Trigger 1 and 2:	Each time a trigger event occurs the counter will increment.
Transit Time:	Set this mode and clear the counter. The counter will begin counting when the next message is transmitted, and stop counting when any message generated by this node is received.
Network Events:	Count incoming network messages.
Free Run @ 26.66 ns:	Increment counter using internal 37.5 MHz clock. Counter will roll over every 1.78 ms.
Free Run @ 1.706 µs with Trigger 2 to CLEAR:	Increment counter using the 585.9 KHz clock. Counter will roll over every 111.8 ms. Assertion of Trigger 2 will clear the counter.

5.10.2 Rollover/Reset

A rollover/reset can generate an interrupt by setting Interrupt On General Purpose Counter/Timer Overflow Mask CSR9[12]. When this bit is set, an interrupt is generated to the host system whenever the counter register (CSR 13) rolls over or overflows. Interrupt On Errors mode CSR0[7] must be enabled in order for this to work properly. The counter/timer will roll over when it reaches 65,535 + 1.

Only one mode may be selected at a time since they use the same counter/timer register (CSR13) for output.

5.10.3 Presetting Values

The counter/timer register counts upward and may be preset with a value to arrive at the desired interrupt interval.

EXAMPLE: To set an interrupt to occur every 100 ms, the counter register is preloaded with '8717', so that when the counter reaches 65,536, only 100 ms would have passed instead of 111.8 ms.

The value of '8717' was determined by dividing the desired interrupt time of 100 ms (100,000 µs) by the increment frequency of 1.760 µs, which results in 56,818. This is the number that would be in the counter register after 100 ms. To obtain a starting value of '8717', subtract 56,818 from 65535. The counter/timer will not roll over until it reaches 65,535 + 1.

See Appendix B, pages B-6, B-12, and B-14 for additional information.

5.11 Modes of Operation

5.11.1 Data Filter

Many implementations of shared-memory tend to rewrite data values to memory that have not actually changed. In order to reduce network traffic, the SCRAMNet+ board has the ability to compare the new value with the old value of data and avoid sending unchanged data values out on the network. This feature is a type of data filtering and can be enabled without affecting node latency while improving network throughput. See Figure 5-4 Data Filter Logic.

CSR0[10] and CSR0[11] control the operation of data filtering as shown in Table 5-6 (see Appendix B for details of CSR operation):

- CSR0[10] enables the data filtering during transmission to the SCRAMNet+ memory and only for the address space above the first 4 K bytes.
- CSR0[11] enables the address space of the first 4 K bytes to be data-filtered in conjunction with CSR0[10].

Table 5-6 Data Filter Options

Bit 11	Bit 10*	Result
0	1	Only the address space above 4 K bytes of shared memory is data-filtered
1	1	All shared memory is data-filtered

* Bit 10 must be ON for any data filtering to take place on that node.

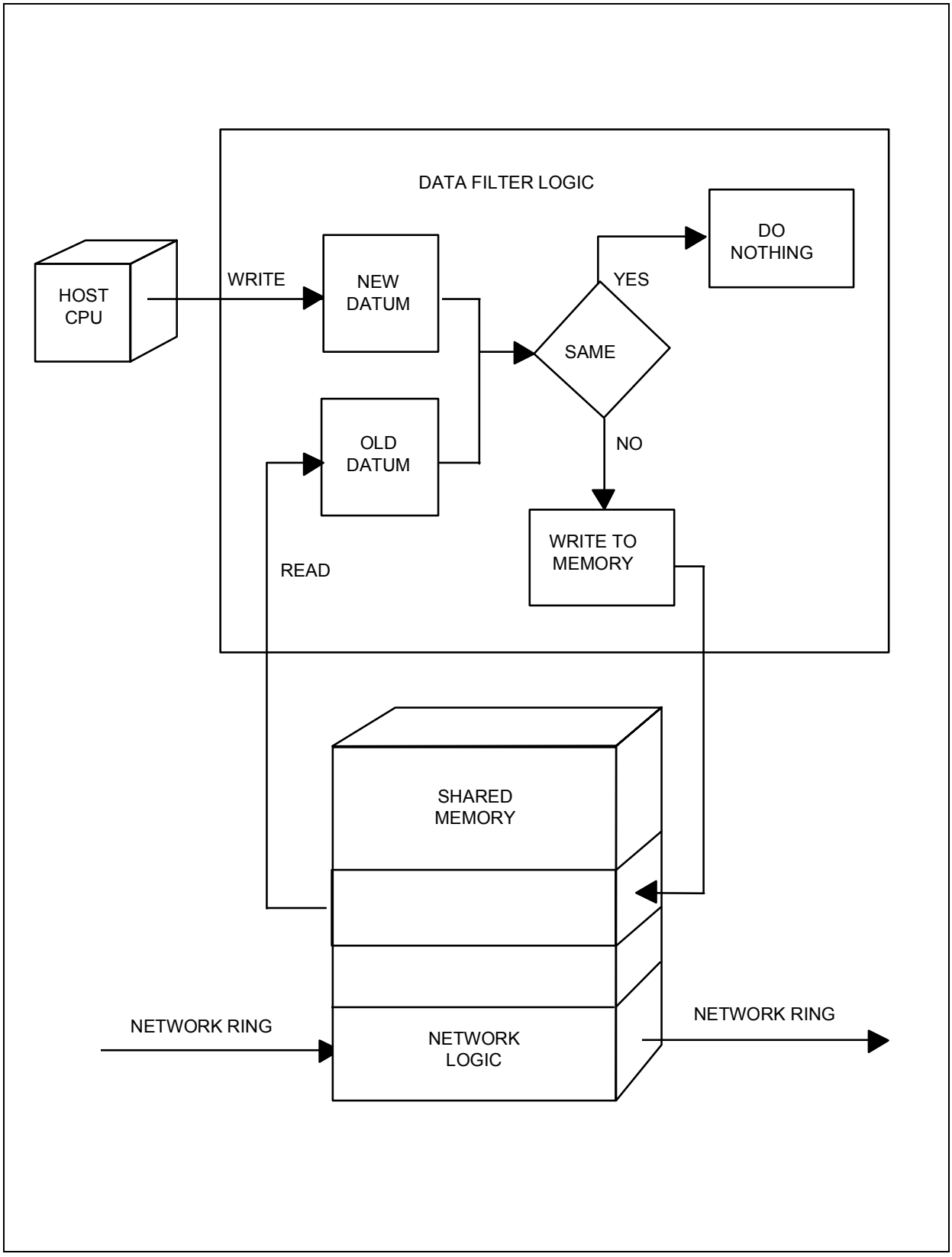


Figure 5-4 Data Filter Logic

5.11.2 HIPRO Mode

WRITE

The SCRAMNet+ network message is based on 32-bit longword data. If a host processor is only capable of 8- or 16-bit data transactions, then the SCRAMNet+ bandwidth is quartered or halved, respectively. For each 32-bit data transaction from the host, two 16-bit data transactions, or four 8-bit transactions will occur on the bus each requiring a SCRAMNet+ network WRITE.

The HIPRO mode was created to provide an efficient means to transmit two 16-bit data transactions as one 32-bit network WRITE. The first of the two 16-bit WRITEs is written to memory but is prevented from going onto the network. The second 16-bit WRITE to memory will trigger the WRITE of the 32-bit location to the network. HIPRO WRITE will not work if Disable Host to Memory Write CSR2[8] is set, because the first 16-bit WRITE must be to the SCRAMNet+ memory.



NOTE: The order of writing the shortwords or bytes into the longword boundary does not matter. However, it is important that a HIPRO location does receive a second shortword WRITE if a first shortword WRITE is initiated, or a total of 4-byte WRITEs if a byte WRITE is initiated, to a HIPRO location. Otherwise, it is possible to partially WRITE a 32-bit location causing the data to be lost and never be transmitted.

The HIPRO mode is also effective for transmitting user-defined 16-bit data items. Two 16-bit data items may be sent as one 32-bit data item if they are consecutive and lie within the same 32-bit address boundary, and interrupts are not being used.

HIPRO mode is selected for those memory addresses which have ACR[4] set. HIPRO Enable CSR2[13] must also be set. Use a non-HIPRO location WRITE to synchronize the HIPRO flags.

READ

The HIPRO READ is controlled by CSR16. This register is CSR enabled and ACR location selectable.

CSR16[0] ON ..... HIPRO READ enabled for every longword address location.

CSR16[1] ON ..... HIPRO READ enabled for all ACR selected HIPRO WRITE locations only. CSR16[0] must be enabled to use this mode.

To effectively conserve host cycles, and in turn increase throughput, HIPRO READ mode allows the host to READ half (1 Shortword or 2 bytes) of the information during the first half access of the data on that longword boundary. On the next host READ operation (not the same location within the same longword boundary) the remaining data is provided without issuing another READ to shared-memory.

5.11.3 Loopback Modes

Each node has a Monitor and Bypass mode, Wire Loopback mode, Mechanical Switch Loopback mode, and a Fiber-optic Loopback mode. These modes are used to check the node's performance and to test transmit/receive circuitry. The loopback mode routes data that would normally be transmitted on to the network directly back to the node from different points.

Table 5-7 depicts the data path for the Monitor and Bypass mode.

Table 5-8 depicts the data path for Wire Loopback Mode.

Table 5-9 depicts the data path for Mechanical Switch Loopback Mode.

Table 5-10 depicts the data path for the Fiber-optic Loopback Mode.

Table 5-11 depicts the data path for the Insert Mode.

MONITOR AND BYPASS MODE

This mode permits the node to receive data only. Network data is not re-transmitted.

Table 5-7 Monitor and Bypass Mode States

State	Register	Setting
Receive Enable	CSR0[0]	ON
Transmit Enable	CSR0[1]	DON'T CARE
Insert Enable	CSR0[15]	OFF
Enable Wire Loopback	CSR2[7]	OFF

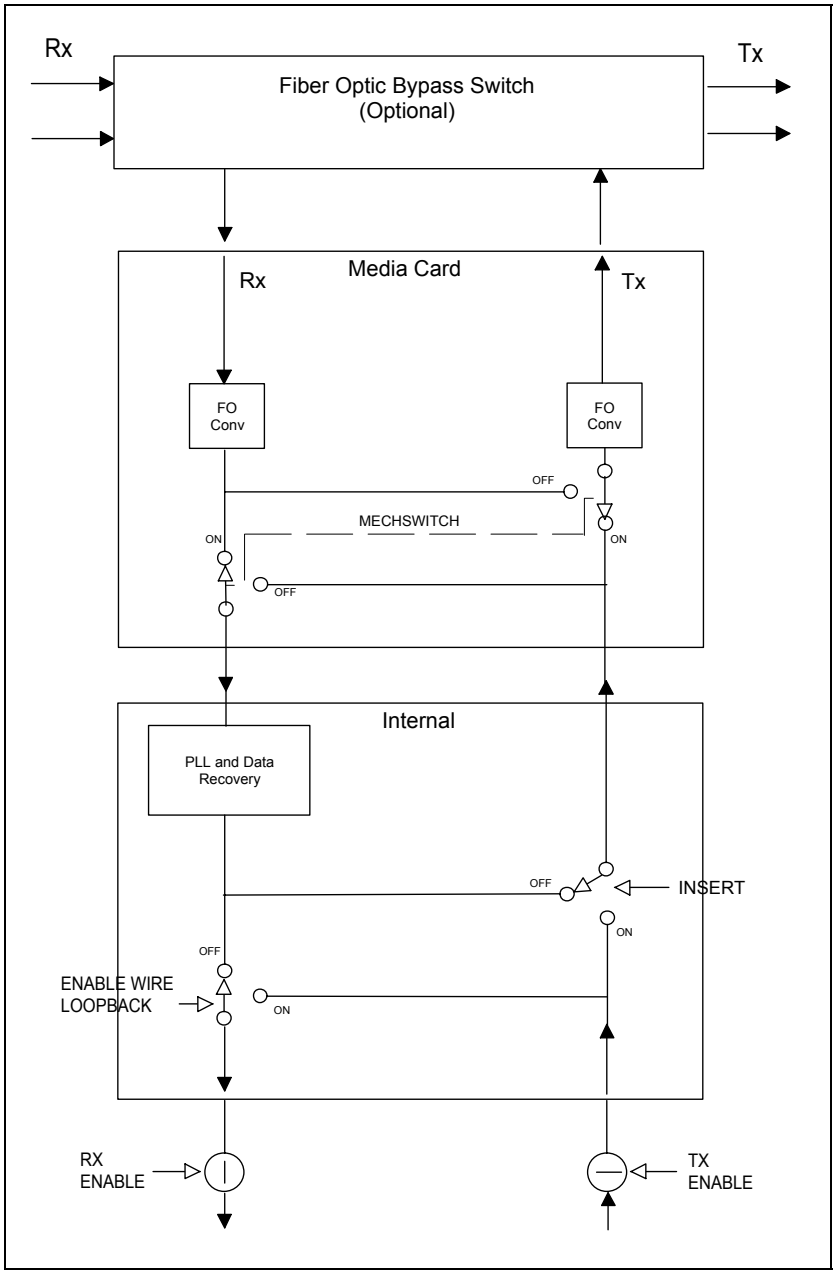


Figure 5-5 Monitor and Bypass Mode

WIRE LOOPBACK MODE

The Wire loopback permits testing of the internal circuitry and needs no manual external modifications to work. In this mode, the transmitted signal does not leave the board.

Table 5-8 Wire Loopback Mode States

State	Register	Setting
Receive Enable	CSR0[0]	ON
Transmit Enable	CSR0[1]	ON
Insert Enable	CSR0[15]	OFF
Enable Wire Loopback	CSR2[7]	ON

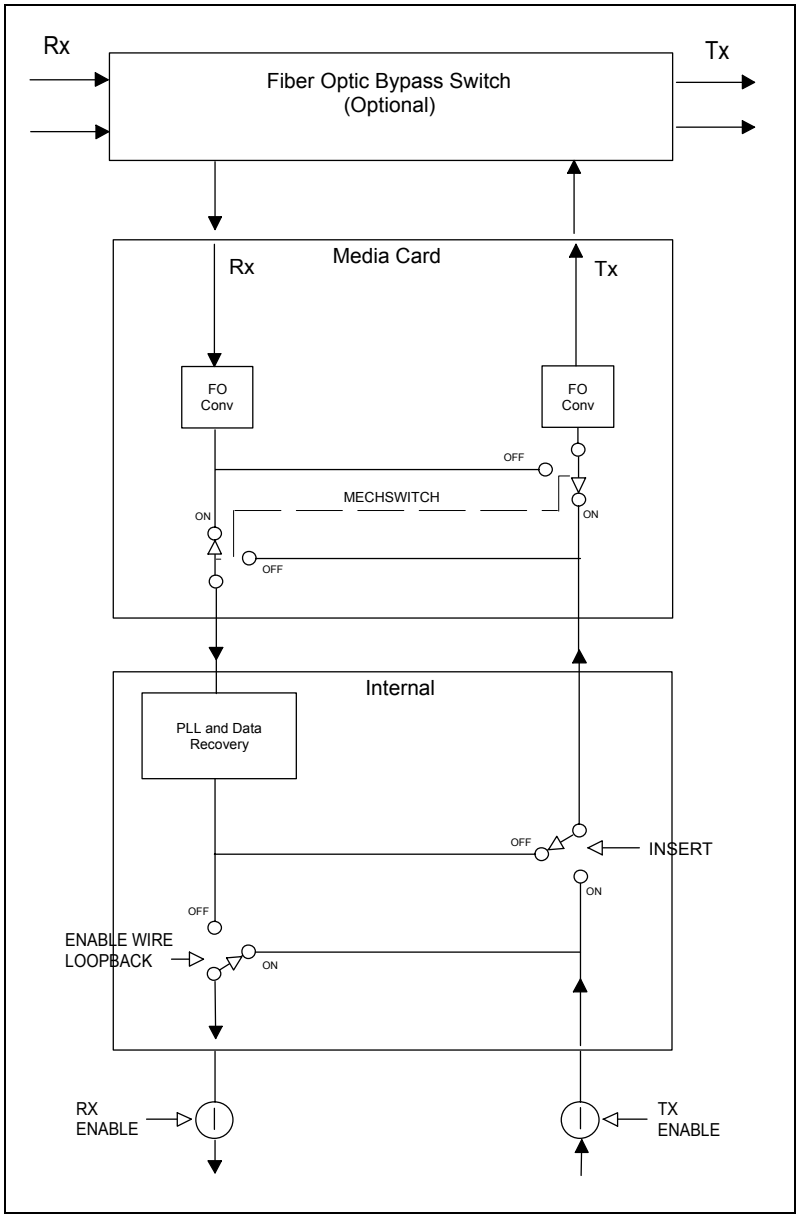


Figure 5-6 Wire Loopback Mode

MECHANICAL SWITCH LOOPBACK MODE

This mode permits testing of all circuitry up to and including a major portion of the Media Card.

Table 5-9 Mechanical Switch Loopback Mode States

State	Register	Setting
Receive Enable	CSR0[0]	ON
Transmit Enable	CSR0[1]	ON
Insert Enable	CSR0[1]5	ON
Enable Wire Loopback	CSR2[7]	OFF
Mechanical Switch Override	CSR8[11]	OFF

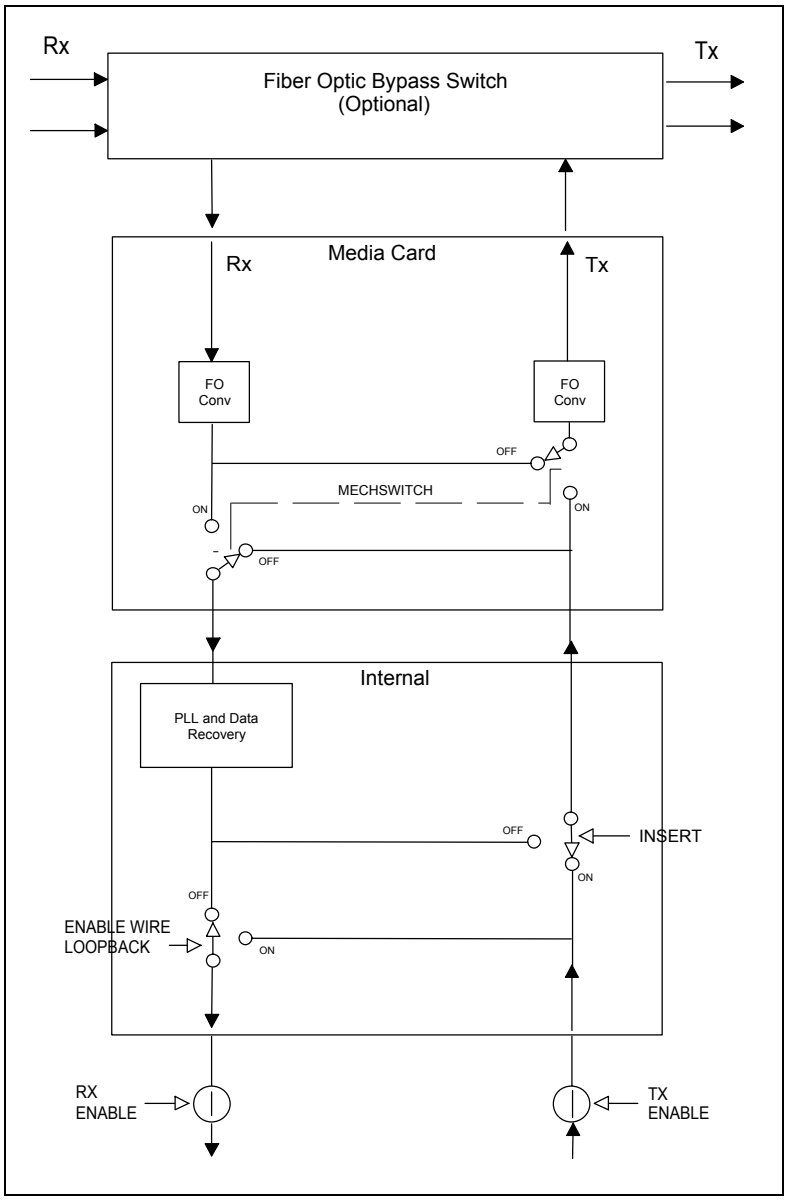


Figure 5-7 Mechanical Switch Loopback Mode

FIBER-OPTIC LOOPBACK

When this mode is invoked, the output of the transmitter is connected by fiber optics directly to the input of the receiver, and the receiver is disconnected from the network.

Table 5-10 Fiber-optic Loopback Mode States

State	Register	Setting
Receive Enable	CSR0[0]	ON
Transmit Enable	CSR0[1]	ON
Insert Enable	CSR0[15]	ON
Enable Wire Loopback	CSR2[7]	OFF
Disable Fiber-optic Loopback	CSR2[6]	OFF
Mechanical Switch Override	CSR8[11]	ON

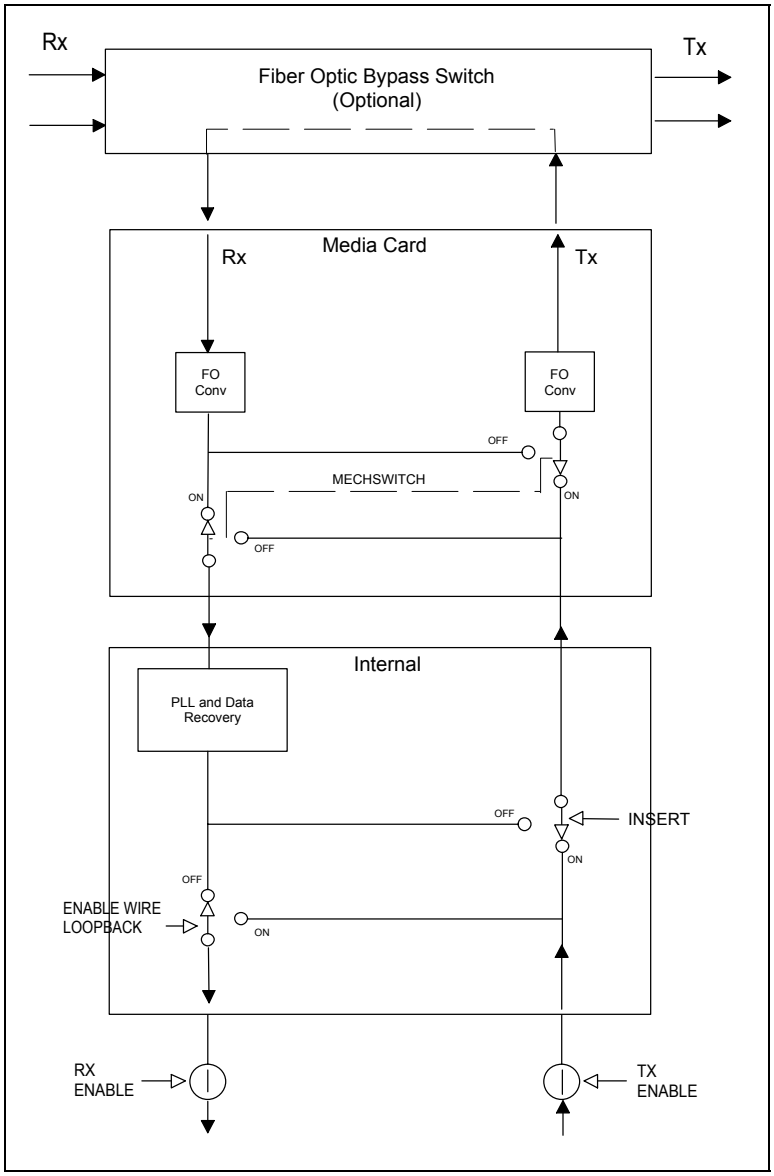


Figure 5-8 Fiber-optic Loopback Mode

The optional Fiber Optic Bypass Switch must be installed for this to work. However, in the absence of the Fiber Optic Bypass Switch, fiber-optic cables could be run from the node's transmitter output connectors to the receiver input connectors. This configuration, with Insert Node enabled, constitutes a Fiber-optic Loopback mode for stand-alone testing. Disable Fiber-optic Loopback CSR2[6] must be set ON when the node is in use as a part of the network. However, this configuration could not be used in a network ring in the place of an Fiber Optic Bypass Switch because it would cause a break in ring continuity.

The Fiber Optic Bypass Switch provides fiber-optic ring continuity when a node is powered down or in loopback mode. CSR2[6] controls the operation of the Fiber Optic Bypass Switch by enabling or disabling Loopback mode. To disable the Fiber-optic Loopback mode, set CSR2[6] ON. This state allows data to be transmitted and received on the network ring for this node.

When the Fiber-optic Loopback mode is enabled (CSR2[6] OFF), the Fiber Optic Bypass Switch does not allow network data to be received by the node. Likewise, no data can be transmitted by the node into the network ring.

When power is lost to the Fiber Optic Bypass Switch, Fiber-optic Loopback mode is enabled regardless of its prior state in order to maintain ring integrity. This is also the default power-up state.

5.11.4 Node Insert Mode

In this mode the node becomes part of the network (Figure 5-9).

Table 5-11 Node Insert Mode

State	Register	Setting
Receive Enable	CSR0[0]	ON
Transmit Enable	CSR0[1]	ON
Insert Node	CSR0[15]	ON
Enable Wire Loopback	CSR2[7]	OFF
Disable Fiber-optic Loopback	CSR2[6]	ON
Mechanical Switch Override	CSR8[11]	ON



NOTE: Do not enable the Wire Loopback and Fiber-optic Loopback and/or Mechanical Switch loopback modes simultaneously. A node in Wire Loopback mode **and** Insert Node will create a break in the network ring that will disable all nodes.

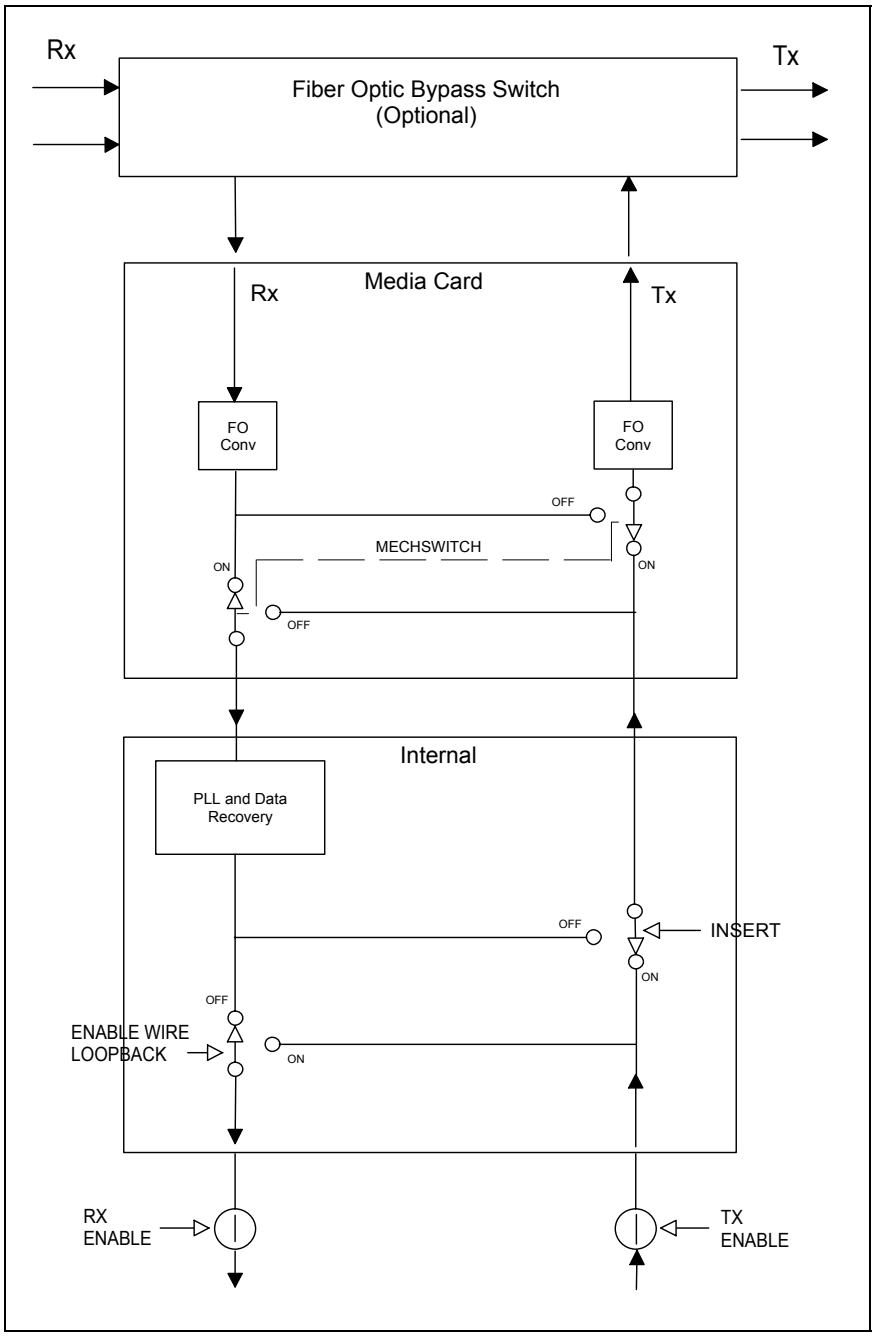


Figure 5-9 Insert Mode

5.11.5 VME Holdoff Mode

To enable VME Holdoff, set CSR8[1] OFF. The VME Holdoff feature automatically slows down CPU data WRITEs to the SCRAMNet+ memory when the Transmit FIFO becomes full. The Transmit FIFO serves as a buffer between the SCRAMNet+ memory and the SCRAMNet+ network.

The Transmit FIFO can become full when the host CPU is writing to SCRAMNet+ memory faster than the network can absorb the data. If a CPU is capable of writing to the SCRAMNet+ memory on the VMEbus at such a rate that the Transmit FIFO becomes full (1024 deep), data could be lost.

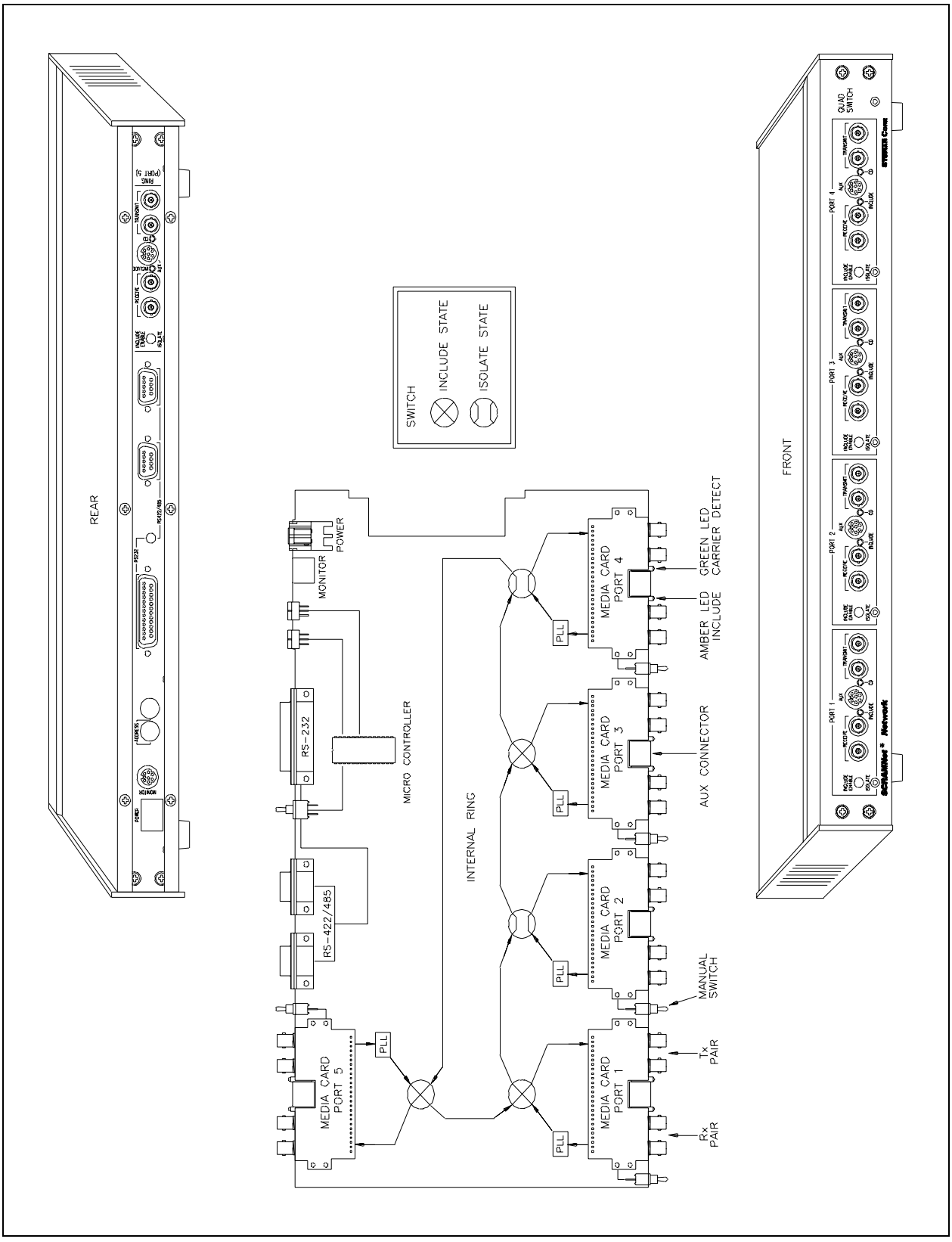


Figure 5-10 Quad Switch

In the event that the Transmit FIFO becomes full, the hardware will automatically extend the next VME WRITE cycle until the Transmit FIFO empties at least one message. This prevents the loss of any data and is transparent to the user.

5.11.6 Write-Me-Last Mode

The Write-Me-Last mode of operation allows the originating node to be the last node in the ring to have the data deposited to its memory. When the host performs a WRITE to the SCRAMNet+ shared-memory, it is not immediately written to the host memory, but is first sent to the other SCRAMNet+ nodes on the network.

Set CSR2[8] and CSR2[9] to enable the Write-Me-Last mode. If desired, this mode can also be used to generate interrupts to the originating node by setting CSR2[10] as well. CSR2[8] is the Disable Host to Shared Memory Write.

5.12 Quad Switch

The Quad Switch is a switching center and is used to dynamically configure active SCRAMNet and SCRAMNet+ ring(s).

The Quad Switch provides dynamic configuration of up to five separate rings. Each separate ring is connected to a port on the Quad Switch. Refer to Figure 5-10. Each ring can be isolated from the other rings or can be included with one or more of the other attached rings.

There is a single logical ring internal to the Quad Switch. The Quad Switch has five external ports, which allow access to this logical ring. Ports 1 through 4 are accessible on the front of the Quad Switch cabinet. Port 5 access is at the rear of the cabinet.

All five ports have standard SCRAMNet transmitters and receivers. Each port can transmit data to and receive data from the internal ring.

The Quad Switch is designed so that a port will be switched into the ring if all its switching controls are enabled. Any one of the switching controls can cause the port to be switched out.

CSR0 IMME and HIE must be set in order to set up interrupts.
CSR1 Bits 0-15 contain various error and status conditions. Interrupts are re-armed whenever any value is written to CSR1.

CSR4 Bits 0-15 contain the interrupt address bits A0-A15.

CSR5 Bits 0-6 contain the interrupt address bits A22-A16.
Bit 15 contains the Interrupt FIFO Not Empty status.

CSR6 Bits 0-7 contain the interrupt vector.

If an interrupt has been received by the host processor from the SCRAMNet+ Network interrupt logic and the appropriate interrupt vector has been initialized in CSR6, then the Interrupt Service Routine linked to that vector will be invoked. Interrupts will be disabled until re-armed by writing to CSR1. Until that time, all other interrupts will be written into the Interrupt FIFO where they can be processed in the Interrupt Service Routine.

If Interrupts on Errors is enabled, then an interrupt due to an error has occurred if the interrupt FIFO is empty on the initial check of CSR5 in the Interrupt Service Routine.

```
Read CSR5
        Test the Interrupt FIFO Not Empty status bit 15
        If (Interrupt FIFO is Empty)
                Read CSR1 to determine the error condition(s)
                Respond to any error conditions
        End if

        While (Interrupt FIFO is NOT Empty)
                Save interrupt address bits A22-A16 from CSR5 (from previous read)
                Read CSR4 and save interrupt address bits A15-A0
                                        • • •
                Service interrupt according to interrupt address data or address
                                        • • •
                Read CSR5 and save Interrupt FIFO Empty status
        Endwhile

        Write to CSR1 to re-enable interrupts
        Return from interrupt service routine
```

Figure 5-11 Interrupt Service Routine

This page intentionally left blank

APPENDIX A

SPECIFICATIONS

TABLE OF CONTENTS

A.1 Hardware Specifications A-1
A.2 Part Number A-2
A.3 Board Dimensions A-3
A.4 VMEbus Voltage Specification A-4
A.5 J8 Connector Pinout A-5
A.6 P2 SCRAMNet+ Pinout (Row A) A-6
A.7 P2 SCRAMNet+ Pinout (Row C) A-7
A.7.1 Fiber Optic Bypass Switch A-8

FIGURES

Figure A-1 VME6U Dimensions A-3
Figure A-2 J8 Connector Pinout A-5
Figure A-3 P2 SCRAMNet+ Pinout - Row A A-6
Figure A-4 P2 SCRAMNet+ Pinout - Row A-7
Figure A-5 Fiber Optic Bypass Switch A-8
Figure A-6 Housing Dimensions A-9

A.1 Hardware Specifications

Hardware Compatibility:	VMEbus ANSI/IEEE std. P1014-1987 VMEbus Manufacturers Group Rev. C.3 Slave device
Physical Dimensions:	
VME6U Card	6.299" x 9.187" 6U Eurocard, one slot
Weight:	
VME6U Card	0.6175 lbs (W/O SIMMs and Media Card, W/face plate)
Media Card, Fiber Optic	0.0915 lbs
SIMM (1)	0.0285 lbs
Electrical Requirements:	+4.75 to +5.25 VDC, 1.5 Amps max. (W/O SIMMs and Media Card)[1]
Temperature Range:	
Storage	-40° to +70°C
Operation	0° to +40°C
Humidity Range:	
Storage	0% to 95% (noncondensing)
Operation	10% to 90% (noncondensing)
Network Line Transmission Rate:	150 million bits/second
Message Length:	
Fixed Length:	82 Bits
Variable Length:	46 bits + 256 or 1024 Data Bytes Maximum
Maximum Nodes on Network Ring:	256
Error Correction:	Available in PLATINUM mode only
Maximum Node Separation:	
Coax:	30 meters
Standard Fiber:	300 meters
Long Link Fiber:	3500 meters
Shared Memory:	
ASIC:	4 KB
On-board	128 KB
Optional Sizes:	
Low Density SIMMs (512 KB)	512 KB, 1 MB, 2 MB
High Density SIMMs (2 MB)	2 MB, 4 MB and 8 MB
Effective Per-Node Bandwidth:	
4 bytes/packet:	6.5 MB/sec
256 bytes/packet:	16.2 MB/sec
1024 bytes/packet:	16.7 MB/sec
Node Latency:	
4 bytes/packet:	250 ns -800 ns
256 bytes/packet:	250 ns - 16 µs
1024 bytes/packet:	250 ns - 61.8 µs
Internal clock speeds:	
SCRAMNet board crystal	150 MHz, +/-100 ppm[2]
26.66 ns timer is a divide-by-four:	37.5 MHz[3]
1.706 µs timer is a divide-by-256:	585.9 KHz[4]

1. Current requirements vary depending on media card type and memory size.
2. Specifications on the crystal demonstrate the precision and stability of the main clock from which all other clocks are derived. This does not include the vagaries introduced by the circuit.
3. The 37.5 MHz clock is the distributed (on board) clock used by other circuits on SCRAMNet host cards.
4. The 585.9 KHz clock is counted by the internal (to ASIC) users timer.

A.2 Part Number

The VME6U adapter part number is in the form:

H-AS-D6VMEL2M-00

where:

CODE	DEFINITION
H	Hardware
AS	Top Level Assembly
D	Standard SCRAMNet+
6VME	6U 16-BIT VME
XXX	**Memory (bytes)**
	04K = 04 K
	128 = 128 K
	512 = 512 K
	L2M = 2 M (LOW DENSITY)
	H2M = 2 M (HIGH DENSITY)
	04M = 4 M
	08M = 8 M
X	**Transmission Media**
	0 = NO Media Card
	1 = COAX Media Card
	2 = STANDARD FO Media Card
	3 = LONGLINK FO Media Card
	4 = LASERLINK FO Media Card
X	**Variable**. Used for product variations and/or modifications

A.3 Board Dimensions

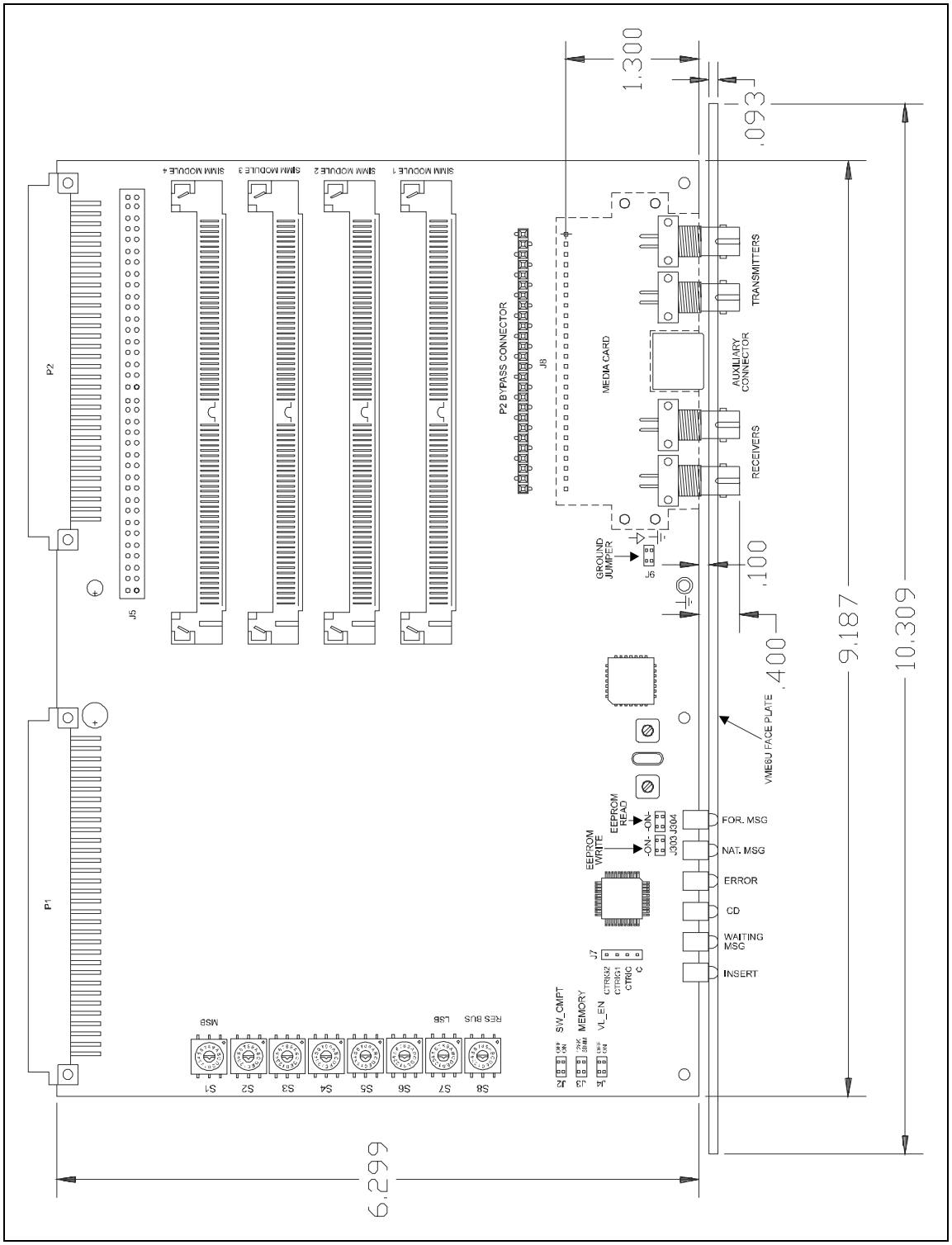


Figure A-1 VME6U Dimensions

A.4 VMEbus Voltage Specification

VMEbus Voltage Specification			
Mnemonic	**Description**	**Allowed Variation**	**Ripple/Noise Below 10 MHz**
+5 V	+5 V dc	+0.25 V/-0.125 V	50 Mv
+12 V *	+12 dc power	+0.60 V/-0.36 V	50 Mv
-12 V *	-12 dc power	-0.60 V/+0.36 V	50 Mv
+5 V STDBY *	+5 V dc standby	+0.25 V/-0.125 V	50 Mv
GND	Ground	Reference	--

* Not used by SCRAMNet+ VME6U

A.5 J8 Connector Pinout

The Media Card is removed, and an adapter is installed connecting J302 and J8. This results in transferring the signals normally going to the Media Card to the J5 connector. When the four 9-pin headers are installed on J5, the signals pass out the P2 connector.

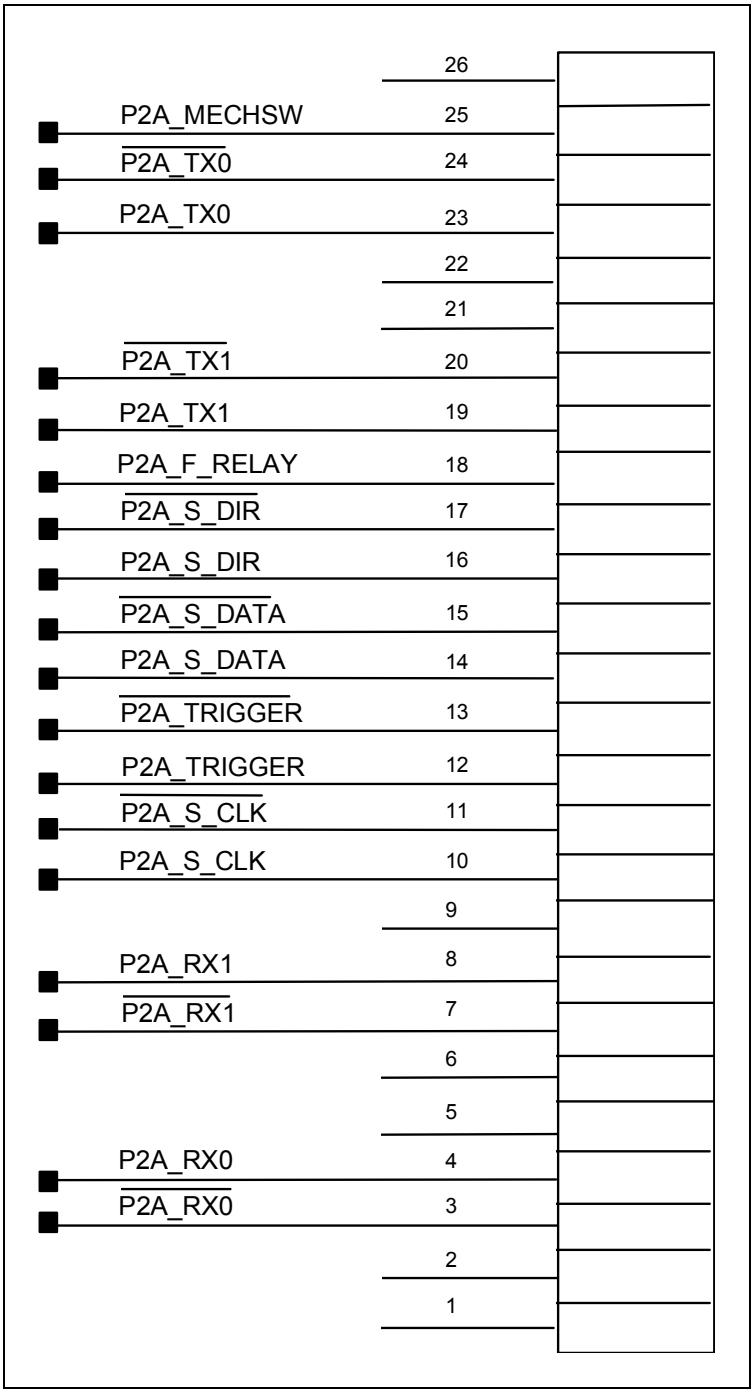


Figure A-2 J8 Connector Pinout

A.6 P2 SCRAMNet+ Pinout (Row A)

NOTE: Row A is active only when the J5 jumpers are installed for the P2 Cabinet Kit.

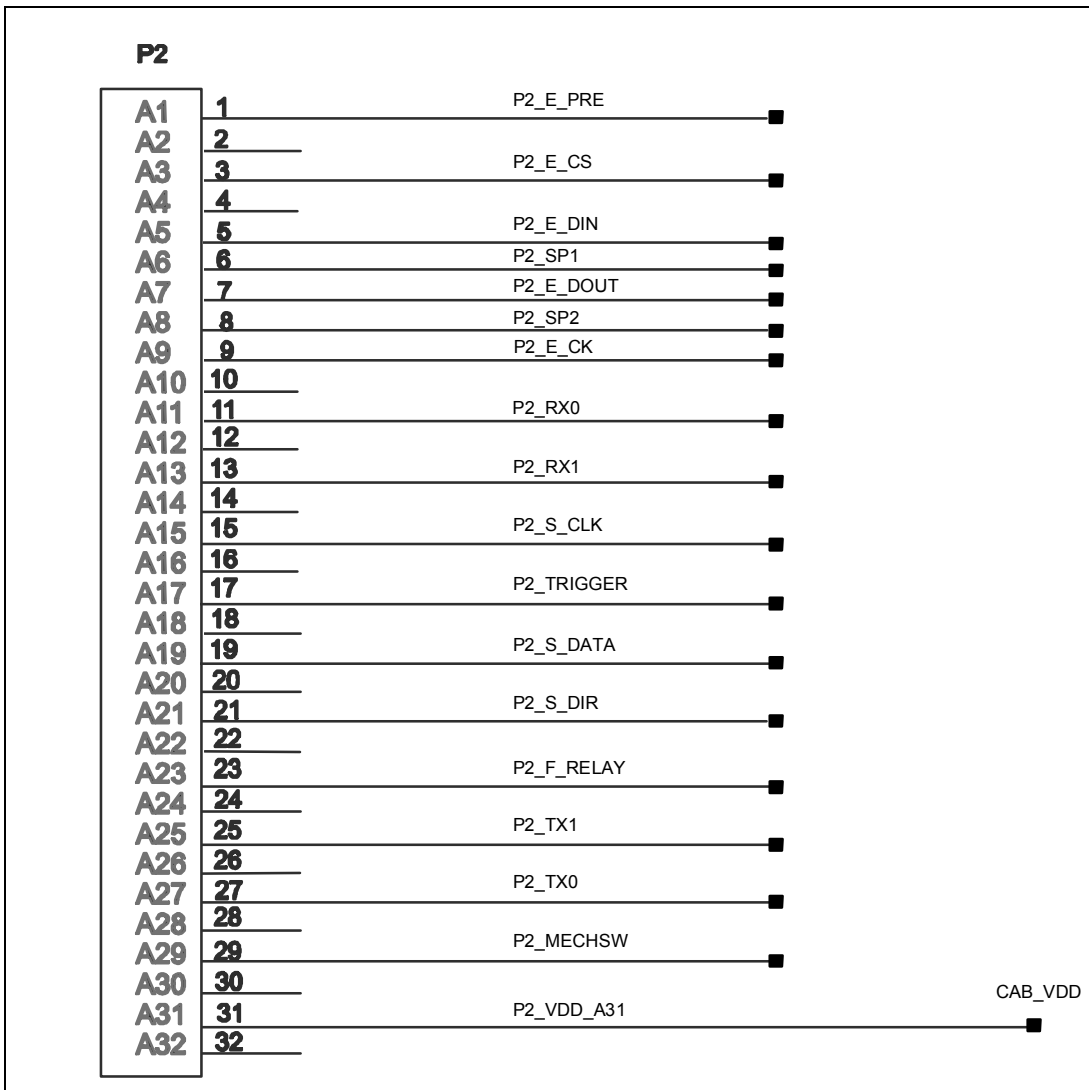


Figure A-3 P2 SCRAMNet+ Pinout - Row A

A.7 P2 SCRAMNet+ Pinout (Row C)

NOTE: Row B is not shown.

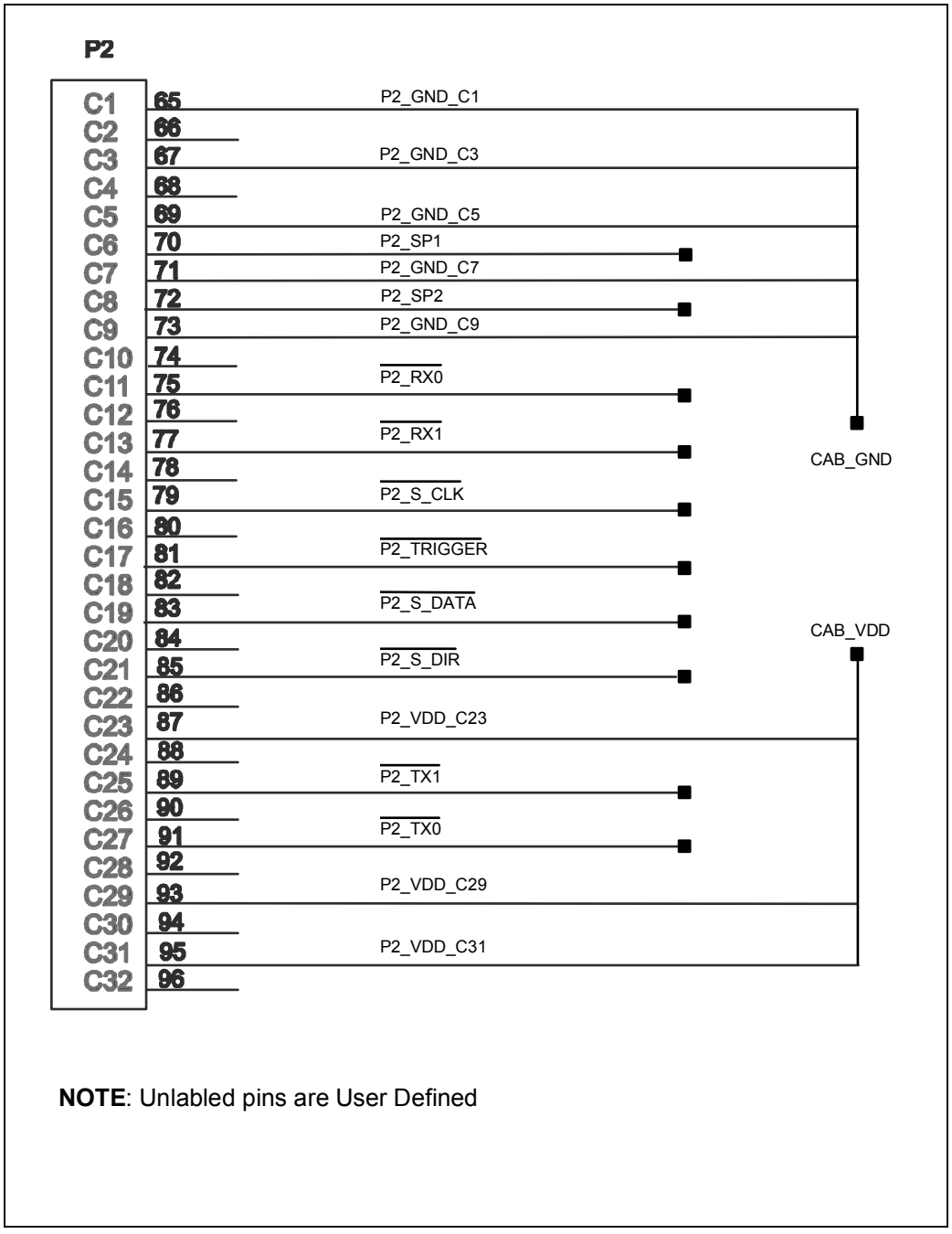


Figure A-4 P2 SCRAMNet+ Pinout - Row

A.7.1 Fiber Optic Bypass Switch

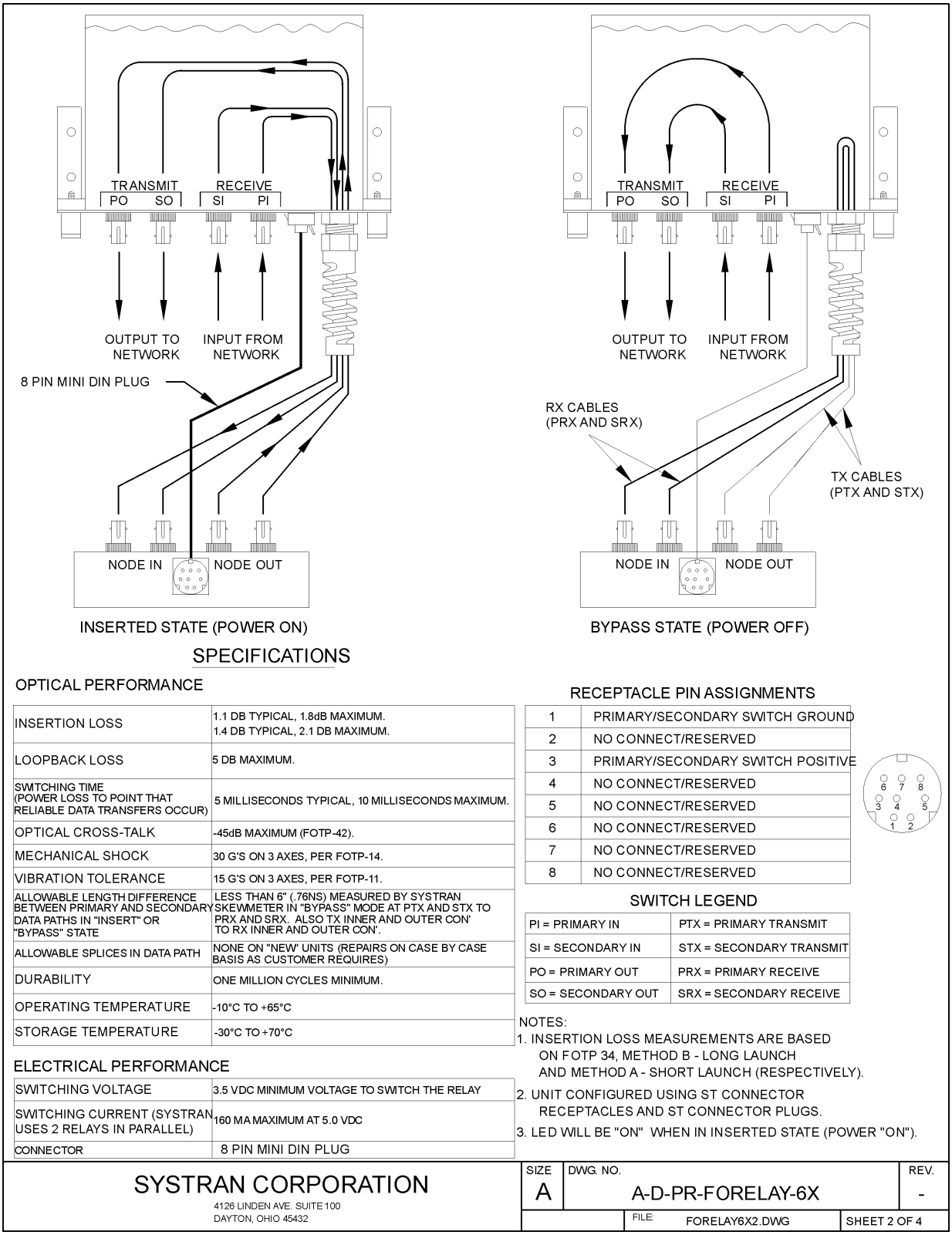


Figure A-5 Fiber Optic Bypass Switch

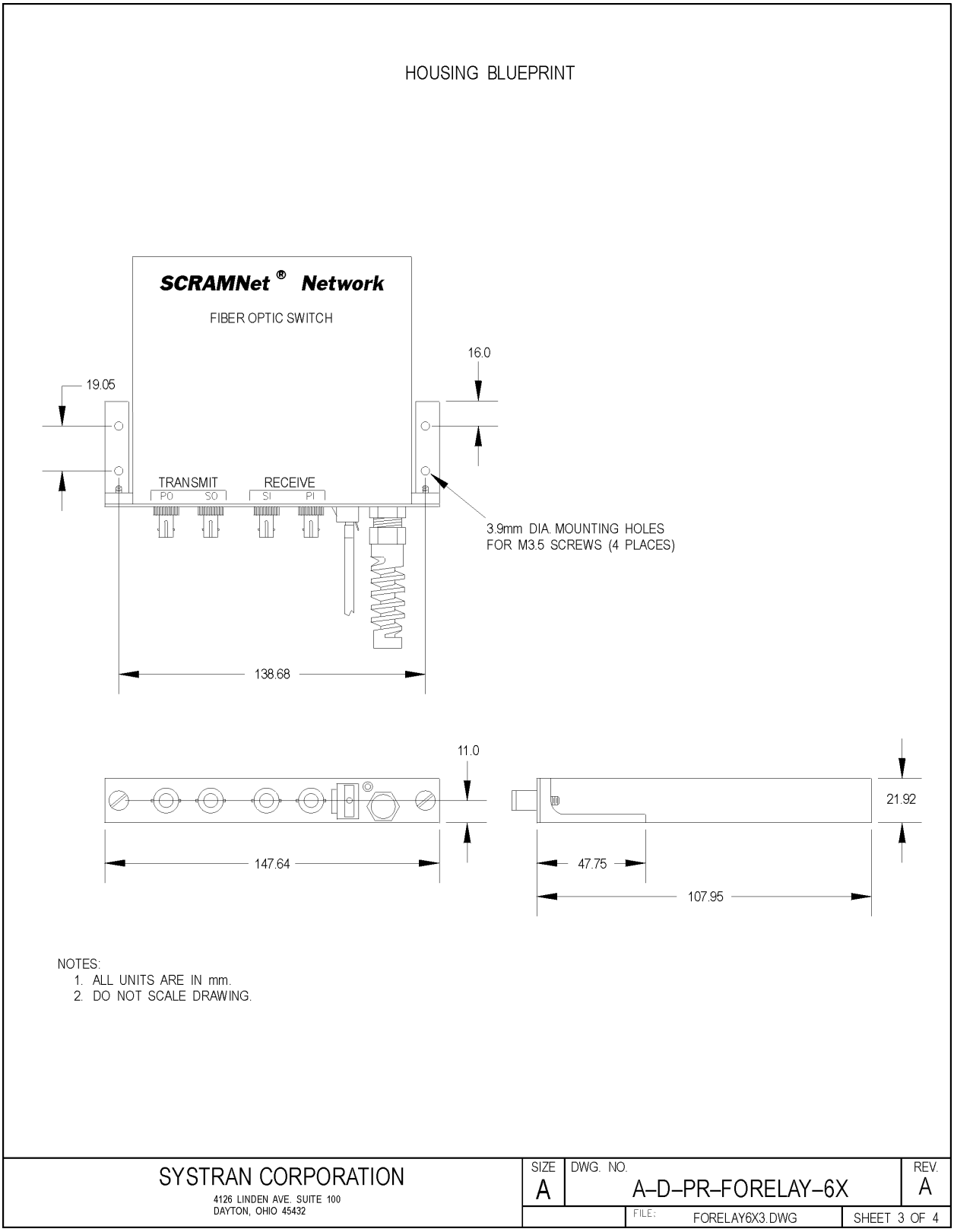


Figure A-6 Housing Dimensions

This page intentionally left blank

APPENDIX B

CSR DESCRIPTIONS

TABLE OF CONTENTS

Table B-1 CSR0 - General SCRAMNet+ Enable and Reset B-2
Table B-2 CSR1 - Error Indicators B-5
Table B-3 CSR2 - Node Control B-7
Table B-4 CSR3 - Node Information B-9
Table B-5 CSR4 - Interrupt Address (LSP) B-9
Table B-6 CSR5 - Interrupt Address and Status (MSP) B-9
Table B-7 CSR6 - Interrupt Vector (Memory Update) B-10
Table B-8 CSR7 - Interrupt Vector (SCRAMNet+ Error) B-10
Table B-9 CSR8 - General SCRAMNet+ Extended Control Register B-11
Table B-10 CSR9 - SCRAMNet+ Interrupt On Error Mask B-12
Table B-11 CSR10 - SCRAMNet+ Shared Memory Address (LSW) B-13
Table B-12 CSR11 - SCRAMNet+ Shared Memory Address (MSW) B-13
Table B-13 CSR12 - Virtual Paging Register B-14
Table B-14 CSR13 - General Purpose Counter/Timer B-15
Table B-15 CSR14 - Reserved B-16
Table B-16 CSR15 -VME Interrupt Priority Level (IRQ) B-16
Table B-17 CSR16 -HIPRO READ Control Bits Register B-17

B.1 Description

This section describes each Control/Status Register and the function of each bit. The name of each bit is indicative of its set state.

The registers are described using bit 0 as the Least Significant Bit (LSB). For example: Inserting 0xA7C3 in a 16-bit register would set bits 0, 1, 6, 7, 8, 9, 10, 13, and 15 ON.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	1	0	0	1	1	1	1	1	0	0	0	0	1	1
A				7				C				3			

Table B-1 CSR0	
Bits	**General SCRAMNet+ Enable and Reset** (READ/WRITE)
1-0	**Network Communications Mode** - Bit 0 controls the receive enable, and Bit 1 controls the transmit enable.
	00 **None** - In this mode, all communications between the node shared memory and the network is inhibited. The node is still able to pass network traffic but does not receive or transmit any data. Loopback modes are also meaningless unless the Host to Shared Memory WRITE bit is enabled.
	01 **Receive Only** - In this mode, any received message is processed and written to node shared memory. Data written by the host is placed in the node shared memory and in the Transmit FIFO but is not sent out on the network. In this mode, the Transmit FIFO will fill and if the Error Interrupt is enabled, Transmit FIFO full interrupt will be triggered. Before changing modes from Receive-Only to either Transmit-Only or Transmit/Receive, the Transmit FIFO should be cleared. If not, all buffered transmit messages will be sent out on the network.
	10 **Transmit Only** - In this mode, any received message bypasses the shared memory and is passed on. Any message written by the host to node shared memory is transmitted on the network. However, any received message is not written to node shared memory. (Transmissions are subject to data filter characteristics.)
	11 **Transmit/Receive** - In this mode, any received message is processed, written to node shared memory and passed on. Any message written by the host to node shared memory is transmitted on the network. This is the normal operation. (Transmissions are subject to data filter characteristics.)
2	**Redundant Transceiver Toggle** - When this bit is cycled ‘0’, ‘1’, ‘0’, the optional redundant transceiver selected link is changed.
3	**Host Interrupt Enable** - When this bit is set, a received message that is written to node shared memory as an interrupt will generate an interrupt request, and the address will be written to the Interrupt FIFO. This bit must be set in order to receive any interrupts from the network.

Table B-1 CSR0 (Continued)	
Bits	**General SCRAMNet+ Enable and Reset** (READ/WRITE)
4	**Auxiliary Control RAM Enable** - When this bit is set, the ACR bytes are swapped in place of the corresponding least-significant byte of every four-byte word in SCRAMNet+ memory. The values written to those ACR byte locations will dictate the type of interrupt that will occur when the 4-byte memory location is written into. The ACR has five bits for interrupt control. They are as follows:
	ACR bit 0 - **Receive Interrupt Enable** - Setting this bit generates an interrupt to the host for network interrupt data received in this location.
	ACR bit 1 - **Transmit Interrupt Enable** - Setting this bit generates an interrupt to the network for a host WRITE to this shared memory location.
	ACR bit 2 - **External Trigger 1** - Setting this bit generates a trigger signal to an external connector whenever there is a host READ/WRITE access to this shared memory location.
	ACR bit 3 - **External Trigger 2** - Setting this bit generates a trigger signal to an external connector whenever there is a network WRITE to this shared memory location.
	ACR bit 4 - **HIPRO location enable** - Setting this bit causes the two 16-bit data or four 8-bit items within the 32-bit address boundary to be transmitted as one 32-bit network message. CSR2[13] must also be set for this action to occur.
5	**Interrupt On Memory Mask Match Enable** - This bit must be set in order for any type of memory interrupt to occur.
6	**Override Receive Interrupt Enable Flag** - When this bit is set, an interrupt is generated to the host by any interrupt data received from the network regardless of the status of the ACR Receive Interrupt bit.
7	**Enable Interrupt on Error** - When this bit is set, Interrupt FIFO Full, Protocol Violation, Bad Message and/or Receiver Overflow conditions causes an interrupt request.
8	**Network Interrupt Enable** - This bit must be set to transmit interrupt data to the network.
9	**Override Transmit Interrupt Enable Flag** - When this bit is set, an interrupt is sent out on the network regardless of the status of the ACR Transmit Interrupt bit.

Table B-1 CSR0 (Continued)	
Bits	**General SCRAMNet+ Enable and Reset** (READ/WRITE)
10	**Enable Transmit Data Filter** - When clear, the entire address space is not filtered and the node is capable of transmitting all messages written to the node shared memory by the host on the network. When set, the data-filter function is enabled for the address space above the first 4 K bytes of SCRAMNet+ memory. Bit 11 controls the lower 4 K bytes.
11	**Enable Lower 4 K Bytes For Data Filter** - When set, the lower 4 K bytes of address space is data filtered if bit 10 is also set. When disabled, the address space will not be filtered.
12	**Reset Receive/Transmit FIFO** - This bit must be toggled from '0' to '1' and back to '0' in order to reset the Receive/Transmit FIFO. The R/T FIFO is a temporary high-speed holding area for data flowing through the network. **NOTE**: If the R/T FIFO were to be reset during active network transmissions, the data in the FIFO at that time would be lost and it would cause errors on the downstream nodes in the network ring.
13	**Reset Interrupt FIFO** - This bit must be toggled from '0' to '1' and back to '0' to reset the Interrupt FIFO.
14	**Reset Transmit FIFO** - This bit must be toggled from '0' to '1' and back to '0' to reset the Transmit FIFO.
15	**Insert Node** - This bit controls the nodes communications mode on the network as either a receiver only or a receiver/transmitter. On power-up, this bit is OFF which translates to the receiver-only mode. This allows user-written software (on each host processor on the network) to be initiated from one node whenever the network is started cold. When this bit is ON, the node is "inserted" into the network ring as a receiver/transmitter which is the normal operating mode if the Fiber Optic Loopback CSR2[6] is disabled. This bit is invalid when the Enable Wire Loopback CSR2[7] is ON.

Table B-2 CSR1	
Bits	**Error Indicators** (READ Only with WRITE/RESET for interrupts) Reading CSR1 will reset the latched error conditions by clearing bits 0,2,4,6,7,8,9,10,11,12,13.
0	**Transmit FIFO Full (Latched)** - When this bit is set, the Transmit-FIFO-Full condition exists. This occurs when there is more data coming from the host to the network than the network can absorb. When the shared memory is full, host WRITEs will be held off by the SCRAMNet+ host interface logic until the Transmit FIFO is no longer full.
1	**Transmit FIFO Not Empty** - This bit does not represent any type of error condition, but rather just a report on the state of the Transmit FIFO. A '0' represents an empty FIFO, where a '1' indicates at least one message in the FIFO.
2	**Transmit FIFO 7/8 Full (Latched)** - This bit indicates that the Transmit FIFO is 7/8 full. A '0' represents a FIFO that is less than 7/8 full, where a '1' indicates the FIFO is backing up and is more than 7/8 full.
3	Always 0
4	**Interrupt FIFO Full (Latched)** - When this bit is set, the Interrupt FIFO Full error condition exists. Reset the Interrupt FIFO by toggling CSR0[13] to ON then to OFF.
5	**Protocol Violation (Latched)** - When this bit is ON, there has been a signal error at the physical layer (fiber or coax) resulting from noise on the transmission lines or a result of hardware failure. It can be any one of the following: Missing transition for two clock periods on line, Parity error or a Framing error.
6	**Carrier Detect (Latched)** - This bit is set if the receivers do not detect any or enough output from the previous nodes transmitters. This is usually an indication that the fiber optic lines have become disconnected or there may be dust/dirt where the fiber optic connections have been made. A visual inspection of the network lines will need to be made.
7	**Bad Message (Latched)** - When this bit is set, the hardware has detected an error in the message packet received on the network. If this error persists, it is an indication that a hardware problem on the SCRAMNet+ board may exist.
8	**Receiver Overflow (Latched)** - When this bit is set, the Receive FIFO has received more data than the node is able to process. This condition may indicate a hardware problem on the board.
9	**Transmit Retry (Latched)** - This bit is set if a message returns to the originating node with bit errors. The message is automatically retransmitted indefinitely until it returns without bit errors. This is considered to be an error condition.
10	**Transmit Retry Time-out (Latched)** - This bit is set if a message does not return to the originating node within the time-out value specified in CSR5. The message is automatically retransmitted indefinitely until it returns. This is considered to be an error condition.

Table B-2 CSR1 (continued)	
Bits	**Error Indicators** (READ Only with WRITE/RESET for interrupts)
11	**Redundant Transmit/Receive Fault (Latched)** - This bit is set if the currently selected optional redundant transceiver has faulted and reverted to the other link. The default value is '0'
12	**General Purpose Counter/Timer Overflow (Latched)** - This bit toggles a 16-bit counter/timer. The events to be counted/timed are set using CSR8[9]; CSR9[13]; and CSR9[14]. The output is held in CSR13. The counter/timer can: count errors, count trigger events for triggers 1 and 2, transmit time, network events, free run @ 26.66 ns, and free run @ 1.706 ns with trigger 2 CLEAR.
13	**Current Link (Latched)** - This bit tells which of the optional redundant transceivers is currently selected as the active link. The default value is 1=A.
14	**Interrupts Armed** - During interrupt operation, this bit indicates that the conditions to receive an interrupt are active. If this bit is '0', then the host will receive no interrupts. When CSR1 is written to, then the interrupts-armed bit will return to an active status.
15	**Fiber Optic Bypass Not Connected** - This is a status bit concerning the installation of the optional Fiber Optic Bypass Switch. A '0' in this bit indicates that the bypass switch is installed while a '1' indicates it is not installed. Fiber Optic Loopback mode CSR2[6] is dependent upon the Fiber Optic Bypass Switch being installed.

Table B-3 CSR2	
Bits	**Node Control** (READ/WRITE)
5-0	These bits are related to lines connected through the MUX control port and are available to the host interface. They are not required to connect to anything
6	**Disable Fiber Optic Loopback** - When this bit is '0' (power up default), the output of the transmitter is connected by fiber optics directly to the input of the receiver, and the receiver is disconnected from the network. The optional Fiber Optic Bypass Switch must be installed for this mode to be effective. This mode is valid only when the Insert Node CSR0[15] is ON. Set this bit to disable the loopback mode when the node is in use as a part of the network.
7	**Enable Wire Loopback** - When this bit is set, the output of the transmitter is connected by wire directly to the input of the receiver, and the receiver is disconnected from the network. The purpose of this bit is purely diagnostic. This mode is valid only when the Insert Node CSR0[15] is OFF.
8	**Disable Host to Memory Write** - When this bit is set, the host WRITEs are not written to the host node's shared memory, but are sent out on the network if Transmit CSR0[1] is ON.
9	**Write Own Slot Enable** - When this bit is set, the originating node can receive the message slot (or packet) sent out to the network. This is not the normal procedure but may be used in conjunction with CSR2[10] when it is desired to generate an interrupt to the host, written by the host.
10	**Enable Interrupt On Own Slot** - When this bit is set, a message with the interrupt bit set can be received by the originating node if CSR2[9], is also set. This coupling enables a host processor to interrupt itself (Self Interrupt).
11	**Message Length Limit** - Variable maximum message size: 1024 bytes or 256 byte. It is used in conjunction with CSR2[12], CSR2[14] and CSR2[15] to enable Plus mode communication protocols.

Write-Me-Last/Self-Interrupt Mode Definition

Bit 10	Bit 9	Bit 8	Mode
0	1	1	WRITE ME LAST mode
1	1	0	SELF-INTERRUPT mode
1	1	1	WRITE ME LAST with SELF-INTERRUPT mode

Table B-3 CSR2 (continued)	
Bits	**Node Control** (READ/WRITE)
12	**Variable Length Messages on Network** - When ON, this bit enables variable length messages. It is used in conjunction with CSR2, bits 11, 14 and 15 to enable PLUS mode communication protocols (see below).
13	**HIPRO Enable** - When this bit is set, the two 16-bit shortwords associated with the longword addressed at ACR bit 4, will be transmitted onto the network as one 32-bit longword. The first shortword WRITE will be held until the second shortword WRITE occurs, which results in the 32-bit data value to be written to the network. Exceptions: HIPRO will not work when Disable Host to Memory WRITE CSR2[8] is set. HIPRO will not work when writing two separate shortwords while using interrupts.
14	**Multiple Messages** - This bit allows multiple native messages on the network. It is used in conjunction with CSR2[11], CSR2[12] and CSR2[15] to enable the BURST mode communication protocol (see below).
15	**No Network Error Correction** - This bit is used in conjunction with CSR2[12] and CSR2[14] to enable communication protocols: BURST or PLATINUM mode and the variable length message PLUS (+) mode (see below).

SCRAMNet+ Protocol Mode Definition

Network Mode	CSR2[15]	CSR2[14]	CSR2[12]	CSR2[11]
	No Error Correction	Multiple Message	Variable Length	Message Size Maximum
BURST	1	1	0	NO MEANING
PLATINUM	0	1	0	NO MEANING
BURST+	1	1	1	1=1024, 0=256
PLATINUM+	0	1	1	1=1024, 0=256

Table B-4 CSR3	
Bits	**Node Information** (READ ONLY)
7-0	**Node Number Count** - These bits represent the total number of SCRAMNet+ nodes on the network. This value is dynamically determined by the hardware. The value ranges from 0 to 255 depending upon the number of nodes actually on the network. **Transmit Age**. This field is also used to READ/WRITE the T_AGE[7:0] field. This register reflects this field when the ID_MUX bit in CSR8[0] is set.
15-8	**Node Identification Number** - These bits represent the SCRAMNet+ node identification number. Each node must have a unique identification number from 0 to 255 for each network ring. The NODE ID need not be in sequential order. **Receive ID**. This field is also used to READ/WRITE the RXID[7:0] field. This register reflects this field when the ID_MUX bit in CSR8[0] is set.

Table B-5 CSR4	
Bits	**Interrupt Address (LSP)** (READ ONLY)
15-0	**LSP of the Interrupt Address** - These bits represent the LSP of the interrupt address (A15 - A0). Bits 0 and 1 are always '0' since the addresses are on four-byte boundaries.

Table B-6 CSR5	
Bits	**Interrupt Address and Status** (READ ONLY)*
6-0	**MSP of the Interrupt Address** - These seven bits represent the MSP of the interrupt address (A22 - A16). When coupled with CSR4, this address represents the SCRAMNet+ memory location of the interrupt.
13-7	Reserved.
14	**Retry Interrupt FIFO Bit** - This bit is set when an interrupt message is received that has its message retry bit set. This can be checked in the interrupt service routine to guard against double interrupts from the same message if it happens to be retransmitted.
15	Interrupt FIFO Not Empty - When this bit is clear, the Interrupt FIFO is empty. Do not READ CSR4 when this bit is '0'. When this bit is set, it signals that CSR5 and CSR4 contain a legitimate interrupt address.

* Writing the Transmit Time-out value to CSR5 stores it in shadow memory. Do not set this value to '0'. A value of '0' prevents host-generated data from leaving the Transmit FIFO.

Table B-7 CSR6	
Bits	**Interrupt Vector (Memory Update)** (READ/WRITE)
7-0	**Interrupt Vector -** This host specific register stores the VMEbus interrupt vector for the interrupt generated by a Memory Update. This register must be pre-loaded with the vector before interrupt processing can occur. *
15-8	Reserved.

Table B-8 CSR7	
Bits	**Interrupt Vector (SCRAMNet+ Error)** (READ/WRITE)
7-0	**Interrupt Vector -** This host specific register stores the VMEbus interrupt vector for the interrupt generated by a SCRAMNet+ Error. This register must be pre-loaded with the vector before interrupt processing can occur. *
15-8	Reserved.

* Both Interrupt Vectors are tied to the same Interrupt Request (IRQ) level set in CSR15

Table B-9 CSR8	
Bits	**General SCRAMNet+ Extended Control Register**
0	**ID Multiplex** - When set to 1, CSR3 contains the T_AGE and RXID fields.
1	**Disable Holdoff** - When set, this bit disables the HOLDOFF feature.
7-2	These bits are used for programming the EEPROM.
8	**CSR Reset** - Setting this bit will cause bus errors. On reset, CSRs loads from EEPROM.
9	**General Purpose Counter/Timer Free Run** - Setting this bit causes the GPC to free run at a rate of 37.5 MHz (26.66 ns). This counter mode overrides all other counter mode settings.
10	**Receive Interrupt Override** - When this bit is set, all incoming network messages are treated as interrupt messages.
11	**Mechanical Switch Override** - Normally set to ON. When OFF, Mechanical Switch Loopback Mode is invoked.
14-12	**Memory Size Configuration** - These bits indicate the memory-size code and are used in conjunction with the memory address stored in CSR10 and 11. The memory size is automatically calculated. (See below)
15	Reserved. (Always 1).

Memory Size Configuration

Bit 14	Bit 13	Bit 12	Memory Size
1	1	1	4 KB
1	1	0	128 KB
1	0	1	512 KB
1	0	0	1 MB
0	1	1	2 MB
0	1	0	4 MB
0	0	1	8 MB

Table B-10 CSR9	
Bits	**SCRAMNet+ Interrupt On-Error Mask***
0	Transmit FIFO Full Mask
1	Transmit FIFO not Empty Mask
2	Transmit FIFO 7/8 Full Mask
3	Built In Self Test Stream (BIST) - Internal 82-bit BIST shift register output.
4	Interrupt FIFO Full Mask
5	Protocol Violation Mask
6	Carrier Detect Fail Mask
7	Bad Message Mask
8	Receiver Overflow Mask
9	Transmitter Retry Mask
10	Transmitter Retry Due to Time Out Mask
11	Redundant TX/RX Fault Mask
12	Interrupt on General Purpose Counter/Timer Overflow Mask
13	See Below
14	See Below
15	Fiber Optic Bypass Switch Not Connected Mask

General Purpose Counter/Timer Modes

CSR8[9]	CSR9[14]	CSR9[13]	Counter/Timer Modes
0	0	0	Count Errors
0	0	1	Count Trigs (1&2)
0	1	0	Transit Time
0	1	1	Network Events
1	1	X	Free Run @ 26.66 ns
1	0	1	1.706 µs w/trig 2 CLR

* To enable an On-Error mask, set the bit to '1'.

Table B-11 CSR10		
Bits	**SCRAMNet+ Replicated Shared Memory Address (LSW)**	
0	SMA_ENABLE	**Shared Memory Address Enable**. This bit enables the on-ASIC comparator for shared-memory access.
11-1	-0-	Always zero
12	SMA12	**Shared Memory Address**
13	SMA13	
14	SMA14	
15	SMA15	

Table B-12 CSR11		
Bits	**SCRAMNet+ Replicated Shared Memory Address (MSW)**	
0	SMA16	This is the most significant part of the replicated shared memory.
1	SMA17	
2	SMA18	
3	SMA19	
4	SMA20	
5	SMA21	
6	SMA22	
7	SMA23	
8	SMA24	
9	SMA25	
10	SMA26	
11	SMA27	
12	SMA28	
13	SMA29	
14	SMA30	
15	SMA31	

Table B-13 CSR12		
Bits	**SCRAMNet+ Virtual Paging Register**	
0	VP	**Virtual Paging Enable**. When ON, this bit enables Virtual Paging.
4-1	-0-	Always zero
5	VP_A12	**Virtual Page number.** The significance of this register is dependent on the memory size. (e.g. For 4 MB, only VP_A22 is valid; for 4 KB, VP_A[22:12] are valid.
6	VP_A13	
7	VP_A14	
8	VP_A15	
9	VP_A16	
10	VP_A17	
11	VP_A18	
12	VP_A19	
13	VP_A20	
14	VP_A21	
15	VP_A22	

Table B-14 CSR13		
Bits	**General Purpose Counter/Timer**	
0	RD_COUNT[0]	This is a General Purpose Counter/Timer register. It can be used to count trigger 1 and 2 events, count errors, or other events as programmed by CSR9, bits 13 and 14.
1	RD_COUNT[1]	
2	RD_COUNT[2]	
3	RD_COUNT[3]	
4	RD_COUNT[4]	
5	RD_COUNT[5]	
6	RD_COUNT[6]	
7	RD_COUNT[7]	
8	RD_COUNT[8]	
9	RD_COUNT[9]	
10	RD_COUNT[10]	
11	RD_COUNT[11]	
12	RD_COUNT[12]	
13	RD_COUNT[13]	
14	RD_COUNT[14]	
15	RD_COUNT[15]	

Table B-15 CSR14	
Bits	**Reserved**
15-0	Not Used

Table B-16 CSR15	
Bits	**VME Interrupt Priority Level (IRQ) External Control Status Register**
0	Not Used
7-1	This is a 7-bit wide, host-specific, READ/WRITE register that holds the VME Interrupt Priority Level (IRQ). Bits reflect the Interrupt Priority Level. For example: IPL 5 translates to setting bit 5 (20 *hex*) **Selector Chart** **Level** **Bit** **Hex** 7 6 5 4 3 2 1 1 0 0 0 0 0 0 1 0x2 2 0 0 0 0 0 1 0 0x4 3 0 0 0 0 1 0 0 0x8 4 0 0 0 1 0 0 0 0x10 5 0 0 1 0 0 0 0 0x20 6 0 1 0 0 0 0 0 0x40 7 1 0 0 0 0 0 0 0x80 Only one level is set at a time. Multiple level selection disables the WRITE operation. For example: You cannot WRITE a value of '0x18' (Level 5 and Level 4).
15-8	Reserved.

Table B-17 CSR16	
Bits	**HIPRO READ Control Bits Register (External Control Status Register)**
1-0	This is a 2-bit wide, High Performance (HIPRO) READ Control Bits Register. Only bits 1 and 0 are valid. Bit 1 Bit 0 0 1 HIPRO READ enabled 1 1 HIPRO READ ACR enabled Bit 0 is CSR enabled. HIPRO READ enabled for every longword-address location. This is an override bit. Bit 1 is ACR selectable. HIPRO READ enabled for all ACR HIPRO WRITE ACR[4] locations only. Both bits 0 and 1 must also be enabled for this mode.
15-2	Reserved

This page intentionally left blank

APPENDIX C

CSR SUMMARY

TABLE OF CONTENTS

C.1 CSR0 - General SCRAMNet+ Enable and Reset C-1
C.2 CSR1 - Error Indicators C-2
C.3 CSR2 - Node Control C-3
C.4 CSR3 - Node Information C-4
C.5 CSR4 - Interrupt Address (LSP) C-5
C.6 CSR5 - Interrupt Address (MSP) and Status (READ Only*) C-6
C.7 CSR6 - Interrupt Vector (Memory Update) (R/W) C-6
C.8 CSR7 - Interrupt Vector (SCRAMNet+ Error) (R/W) C-6
C.9 CSR8 - General SCRAMNet+ Extended Control Register C-7
C.10 CSR9 - SCRAMNet+ Interrupt-On-Error Mask C-8
C.11 CSR10 - SCRAMNet+ Shared Memory Address (LSW) C-9
C.12 CSR11 - SCRAMNet+ Shared Memory Address (MSW) C-9
C.13 CSR12 - Virtual Paging Register C-10
C.14 CSR13 - General Purpose Counter /Timer C-10
C.15 CSR14 - Reserved C-11
C.16 CSR15 - VME Interrupt Priority Level (IRQ) C-11
C.17 CSR16 - HIPRO Read Control Bits Register C-11
C.18 Auxiliary Control RAM (R/W) C-11

C.1 CSR0 - General SCRAMNet+ Enable and Reset

Bit	Function	Name
0	Receive Enable	RX_ENB
1	Transmit Enable	TXEN
2	Redundant TxRx Toggle	RTT
3	Host Interrupt Enable	HIE
4	Auxiliary Control RAM Enable	ACRE
5	Interrupt on Memory Mask Match Enable	IMME
6	Override RIE Flag	ORF
7	Interrupt on Errors	IOE
8	Network Interrupt Enable	NIE
9	Override TIE Flag	OTF
10	Enable Tx Data Filter	DFEN
11	Enable Lower 4 Kbytes for Data Filter	EN4K
12	RESET Rx/Tx FIFO	RTRF
13	RESET Interrupt FIFO	RSTIF
14	RESET Transmit FIFO	RTXF
15	Insert Node	INSRT

C.2 CSR1 - Error Indicators

Bit	Function	Name
0	Transmit FIFO Full	TXFF
1	Transmit FIFO Not Empty	TXFNE
2	Transmit FIFO 7/8 Full	TXFAF
3	(Always 0)	Not Used
4	Interrupt FIFO Full	IFF
5	Protocol Violation	PV
6	Carrier Detect Failure	CDF
7	Bad Message	BB
8	Receiver Overflow	RXO
9	Transmit Retry	TXRTY
10	Transmit Retry Time-out	TO
11	Redundant TxRx Fault	RTF
12	General Purpose Counter/Timer Overflow	GPCTO
13	Redundant TxRx Link 1=A/0=B	RTLAB
14	Interrupts Armed – Write to re-arm	IARM
15	Fiber Optic Bypass Not Connected	FOB

C.3 CSR2 - Node Control

Bit	Function	Name
5-0	Available to Host	
6	Disable Fiber Optics Loopback	FO_DIS
7	Enable Wire Loopback	EN_WR_LPB
8	Disable Host to Memory Write	DIS_H_M_WR
9	Enable Write of Our Own Slot to Memory	WOSEN
10	Enable Interrupt on Receipt of Own Interrupt Slot	IOSEN
11	1024 vs 256 variable size max (bytes)	LEN_LIMIT
12	Variable length messages on network	VAR_LEN
13	HIPRO Write Enable	HIPRO
14	Allow multiple native messages on network	MULTIPLE_MSG
15	No Network Error Correction	NO_ERR_CRCT

Write-Me-Last/Self-Interrupt Mode Definition

Bit 10	Bit 9	Bit 8	Mode
0	1	1	WRITE ME LAST mode
1	1	0	SELF-INTERRUPT mode
1	1	1	WRITE ME LAST with SELF-INTERRUPT mode

SCRAMNet+ Protocol Mode Definition

Network Mode	CSR2[15] No Error Correction	CSR2[14] Multiple Message	CSR2[12] Variable Length	CSR2[11] Message Size Maximum
BURST	1	1	0	NO MEANING
PLATINUM	0	1	0	NO MEANING
BURST+	1	1	1	1=1024, 0=256
PLATINUM+	0	1	1	1=1024, 0=256

C.4 CSR3 - Node Information

Bit	Function	Name
0	Node Number Count* (Valid After a Transmission from the Node)	NN0
1		NN1
2		NN2
3		NN3
4		NN4
5		NN5
6		NN6
7		NN7
8	Node ID Number*	NID0
9		NID1
10		NID2
11		NID3
12		NID4
13		NID5
14		NID6
15		NID7

* When ID_MUX bit CSR8[0] is set:
Bits [7:0] are **Transmit Age**
Bits [15:8] are **Receive ID**

C.5 CSR4 - Interrupt Address (LSP)

Bit	Function	Name
0	Interrupt FIFO Address Field (LSP)	Always = 0
1		Always = 0
2		RFA2
3		RFA3
4		RFA4
5		RFA5
6		RFA6
7		RFA7
8		RFA8
9		RFA9
10		RFA10
11		RFA11
12		RFA12
13		RFA13
14		RFA14
15		RFA15

C.6 CSR5 - Interrupt Address (MSP) and Status (READ Only*)

Bit	Function	Name
0	Interrupt FIFO Address Field (MSP)	RFA16
1		RFA17
2		RFA18
3		RFA19
4		RFA20
5		RFA21
6		RFA22
13-7	Reserved	0
14	Retry Bit in Interrupt FIFO	(RF_RETRY)
15	Interrupt FIFO Not Empty	(~RX_F_E)

* Writing the Transmit Time-out value to CSR5 stores it in shadow memory. Do not set this value to '0'. A value of '0' prevents host-generated data from leaving the Transmit FIFO

C.7 CSR6 - Interrupt Vector (Memory Update) (R/W)

An 8-bit register that holds the VMEbus interrupt vector generated by a memory update. Only bits [7:0] are valid; [15:8] are reserved.

C.8 CSR7 - Interrupt Vector (SCRAMNet+ Error) (R/W)

An 8-bit register that holds the VMEbus interrupt vector generated due to a SCRAMNet+ error. Only bits [7:0] are valid; [15:8] are reserved.

C.9 CSR8 - General SCRAMNet+ Extended Control Register

Bit	Function	Name
0	1 is CSR3=T_AGE & RXID fields	ID_MUX
1	Disable HOLDOFF feature	DIS_HOLD
2	Chip select to EEPROM	CSR_CS0
3	Ext. Chip Select for AUX MICROWIRE peripheral	CSR_CS1
4	MICROWIRE DOUT pin	CSR_DOUT
5	EEPROM program enable	E_PRE
6	CLK line to MICROWIRE port	CSR_CK
7	DIN line connected to the MICROWIRE DOUT pins	E_DIN
8	Initiate initiation sequence - CSR Reset	CSR_RST
9	Override Counter mode	GPC_FRE
10	Receive Interrupt Override	RX_INT_OVR
11	1 = Mechanical Switch Override 0 = Invoke Mechanical Switch Loopback Mode	C_MECHSW
12	Memory Size Configuration (See below)	MCI0
13	Memory Size Configuration (See below)	MCI1
14	Memory Size Configuration (See below)	MCI2
15	Reserved (always 1)	1

Memory Size Configuration

Bit 14	Bit 13	Bit 12	Memory Size
1	1	1	4 KB
1	1	0	128 KB
1	0	1	512 KB
1	0	0	1 MB
0	1	1	2 MB
0	1	0	4 MB
0	0	1	8 MB

C.10 CSR9 - SCRAMNet+ Interrupt-On-Error Mask

Bit	Function	Name
0	Transmit FIFO Full mask	M_TX_F_F
1	Transmit FIFO Not Empty mask	M_TX_F_E
2	Transmit FIFO 7/8 Full Mask	M_TX_F_AF
3	Internal 82 bit BIST shift register output	BIST_STREAM
4	Receiver FIFO Full Mask	M_RX_F_F
5	Protocol Violation mask	M_PV
6	Carrier Detect Fail mask	M_CD_FAIL
7	Bad Message mask	M_BM
8	Receiver Overflow mask	M_RX_OVR
9	Transmitter Retry mask	M_RETRY
10	Transmitter Retry - Time-out	M_RETRY_T_O
11	Redundant Transmit/Receive Fault mask	M_FAULT
12	Interrupt on General Purpose Counter/Timer Overflow	M_COUNT_OVR
13	General Purpose Counter Modes (See below)	M_INC_TRIGS
14	General Purpose Counter Modes (See below)	M_INC_ERRS
15	Fiber Optic Bypass Not Connected mask	M_FO_BYPASS

General Purpose Counter/Timer Modes

CSR8[9]	CSR9[14]	CSR9[13]	Counter/Timer Modes
0	0	0	Count Errors
0	0	1	Count Triggers (1 & 2)
0	1	0	Transit Time
0	1	1	Network Events
1	1	X	Free Run @ 26.66 ns
1	0	1	1.706 µs\w Trig 2 CLR

C.11 CSR10 - SCRAMNet+ Shared Memory Address (LSW)

Bit	Function	Name
0	Enable comparator for SM access	SMA_ENABLE
11-1	Reserved	0
12	Shared Memory Address (LSW)	SMA12
13		SMA13
14		SMA14
15		SMA15

C.12 CSR11 - SCRAMNet+ Shared Memory Address (MSW)

Bit	Function	Name
0	Shared Memory Address (MSW)	SMA16
1		SMA17
2		SMA18
3		SMA19
4		SMA20
5		SMA21
6		SMA22
7		SMA23
8		SMA24
9		SMA25
10		SMA26
11		SMA27
12		SMA28
13		SMA29
14		SMA30
15		SMA31

C.13 CSR12 - Virtual Paging Register

(Refer to Chapter 5, Section 5.2.1, and Appendix B, page B-14)

Bit	Function	Name
0	Enables Virtual Paging when set	VP
4-1	Always '0'	0
5	Virtual Page Number	VP_A12
6		VP_A13
7		VP_A14
8		VP_A15
9		VP_A16
10		VP_A17
11		VP_A18
12		VP_A19
13		VP_A20
14		VP_A21
15		VP_A22

C.14 CSR13 - General Purpose Counter /Timer

(Refer to Chapter 5, Section 5.9, and Appendix B, pages B-6, B-12, and B-15)

Bit	Function	Name
0	Counter/Timer register	RD_COUNT[0]
1	Counter/Timer register	RD_COUNT[1]
2	Counter/Timer register	RD_COUNT[2]
3	Counter/Timer register	RD_COUNT[3]
4	Counter/Timer register	RD_COUNT[4]
5	Counter/Timer register	RD_COUNT[5]
6	Counter/Timer register	RD_COUNT[6]
7	Counter/Timer register	RD_COUNT[7]
8	Counter/Timer register	RD_COUNT[8]
9	Counter/Timer register	RD_COUNT[9]
10	Counter/Timer register	RD_COUNT[10]
11	Counter/Timer register	RD_COUNT[11]
12	Counter/Timer register	RD_COUNT[12]
13	Counter/Timer register	RD_COUNT[13]
14	Counter/Timer register	RD_COUNT[14]
15	Counter/Timer register	RD_COUNT[15]

C.15 CSR14 - Reserved

A 16-bit, READ Only Systran reserved register.

C.16 CSR15 - VME Interrupt Priority Level (IRQ)

A 7-bit, READ/WRITE register that holds the VME Interrupt Priority Level (IRQ).

C.17 CSR16 - HIPRO Read Control Bits Register

Bit	Function
1-0	This is a 2-bit wide, High Performance (HIPRO) READ Control Bits Register. Only bits 1 and 0 are valid. Bit 1 Bit 0 0 1 HIPRO READ enabled 1 1 HIPRO READ ACR enabled Bit 0 is CSR enabled. HIPRO READ enabled for every longword-address location. This is an override bit. Bit 1 is ACR selectable. HIPRO READ enabled for all ACR HIPRO WRITE ACR[4] locations only. Both bits 0 and 1 must also be enabled for this mode.
15-2	Reserved

C.18 Auxiliary Control RAM (R/W)

Bit	Function	Name
0	Receive Interrupt Enable	RIE
1	Transmit Interrupt Enable	TIE
2	External Trigger 1 (Host Read/Write)	ET1
3	External Trigger 2 (Network Write)	ET2
4	HIPRO	HIPRO
7-5	Reserved	0

This page intentionally left blank

APPENDIX D

CABINET KIT

TABLE OF CONTENTS

D.1 Options D-1
D.2 Compact Cabinet Kit D-1
D.3 Expanded Cabinet Kit D-2
D.4 Direct-attached P2 Cabinet Kit D-3

FIGURES

Figure D-1 Compact Cabinet Kit Connection D-1
Figure D-2 Expanded Cabinet Kit Connection D-2
Figure D-3 Direct-attach P2 Cabinet Kit Installation D-3

D.1 Options

Systran provides several cabinet kit options. These are described in detail in the *SCRAMNet Network Cabinet Kit Hardware Reference* (Doc. No. D-T-MR-CABKIT).

D.2 Compact Cabinet Kit

The Compact Cabinet Kit for the SCRAMNet+ Network provides fiber-optic or coax cable access to the node's connections and maintains the shielding of the chassis. LED indicators at the bulkhead plate remotely indicate the node's status.

The host interface board configured to work with the cabinet kit has no media card with transmitters or receivers. Instead, a 20-pin connector and cables form an extension to the cabinet kit, and the media card is mounted on the cabinet kit board.

The Compact model as shown in Figure D-1 has one media card mounted on a cabinet kit board that is attached to a faceplate.

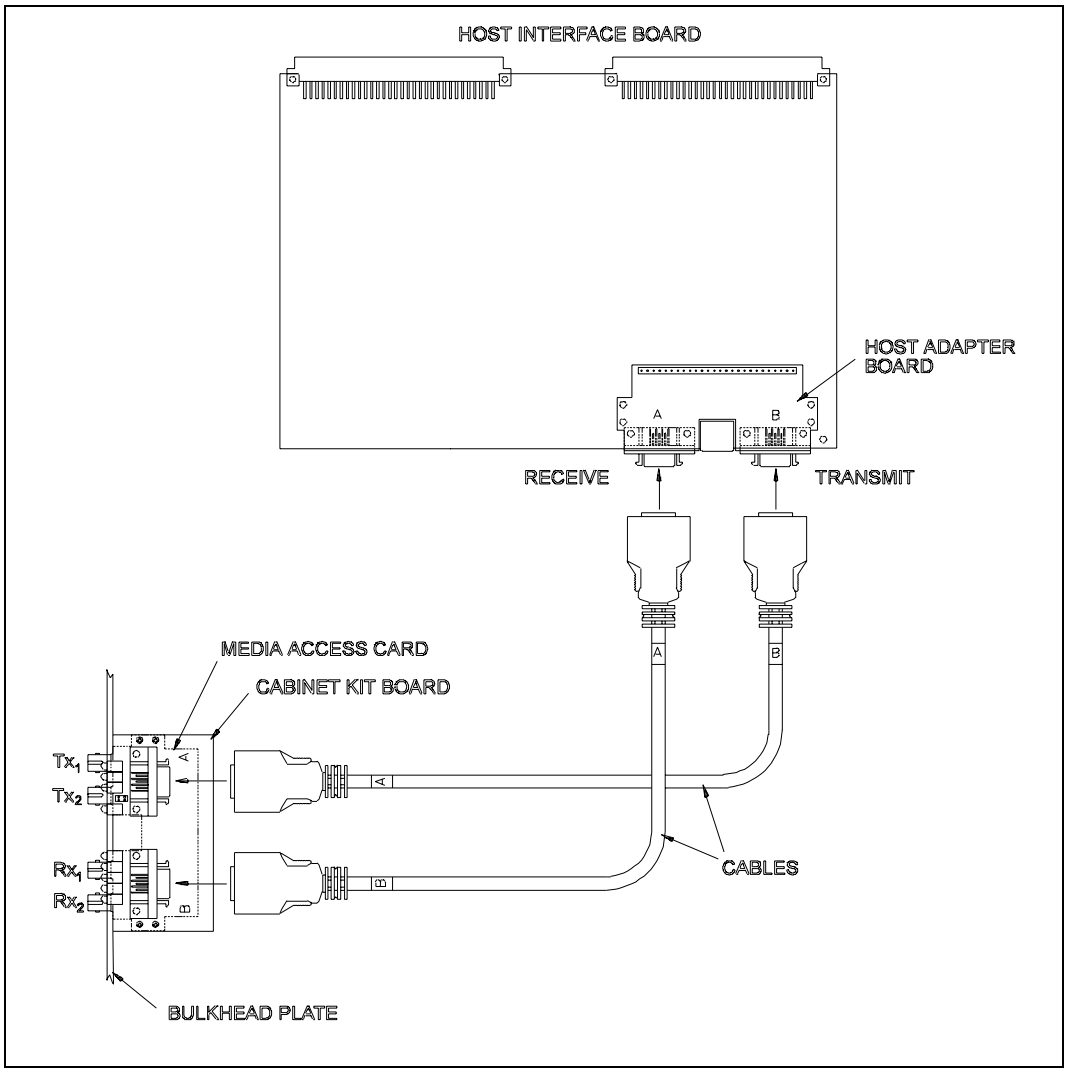


Figure D-1 Compact Cabinet Kit Connection

D.3 Expanded Cabinet Kit

The Expanded model shown in Figure D-2 has connections for up to two media cards providing the option of signal redundancy. This model also comes with a faceplate.

The Expanded cabinet kit has a VME P2 cabling option. This permits connection to the cabinet kit board via the P2 backplane connection.

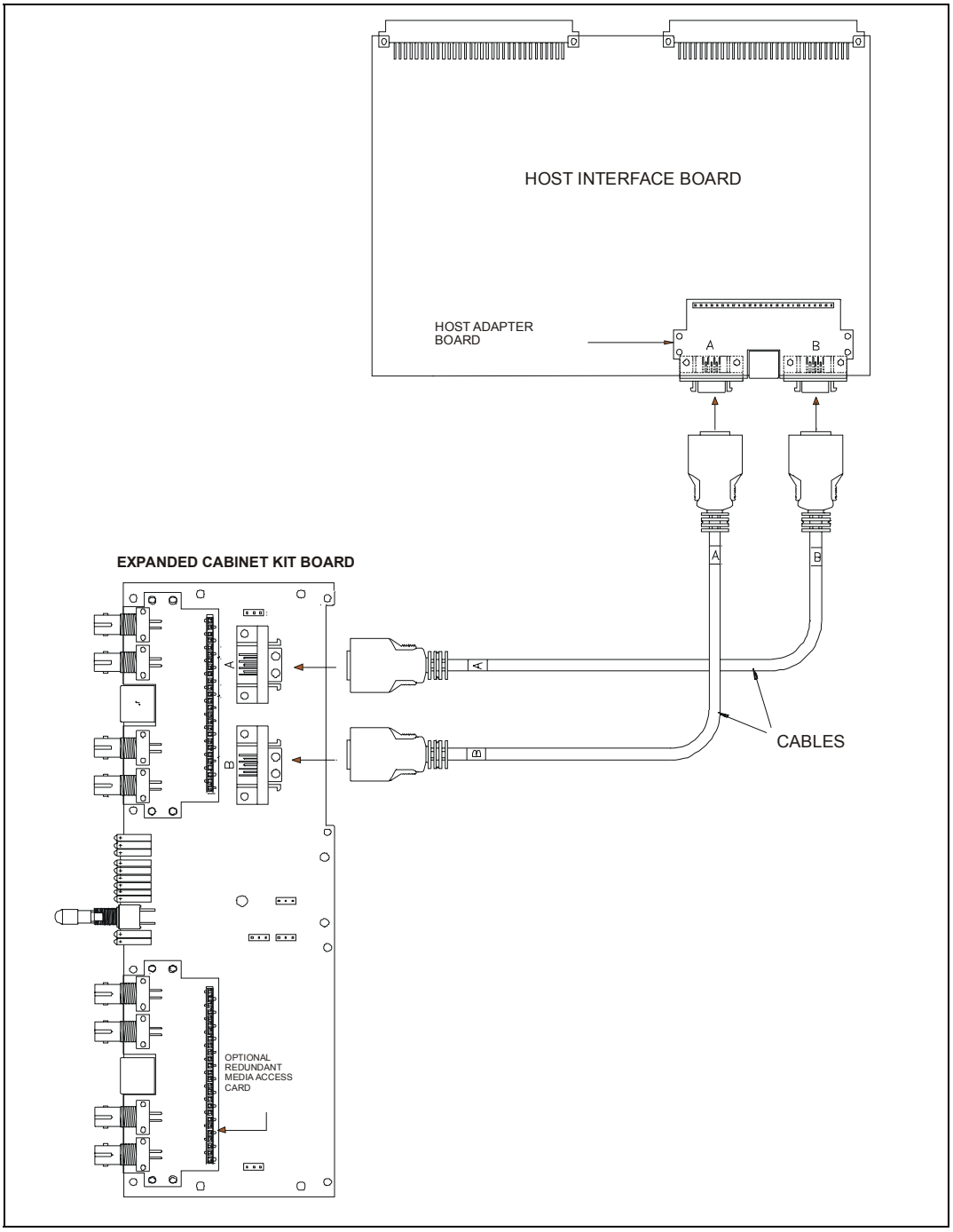


Figure D-2 Expanded Cabinet Kit Connection

Additional details concerning features and installation are included in *SCRAMNet Network Cabinet Kit Hardware Reference*.

D.4 Direct-attached P2 Cabinet Kit

The Direct-Attached P2 cabinet kit provides an interface with the SCRAMNet+ host board via the P2 connector on the back of the VME6U board. It attaches directly to the connector with no intermediate cable. The kit consists of a cabinet-kit board and a DIN 41612-connector shell and screws to adapt the backside of the VME backplane for connection to this board. There is a single connector on the board for a fiber-optic or coax media card; or a host-adapter board

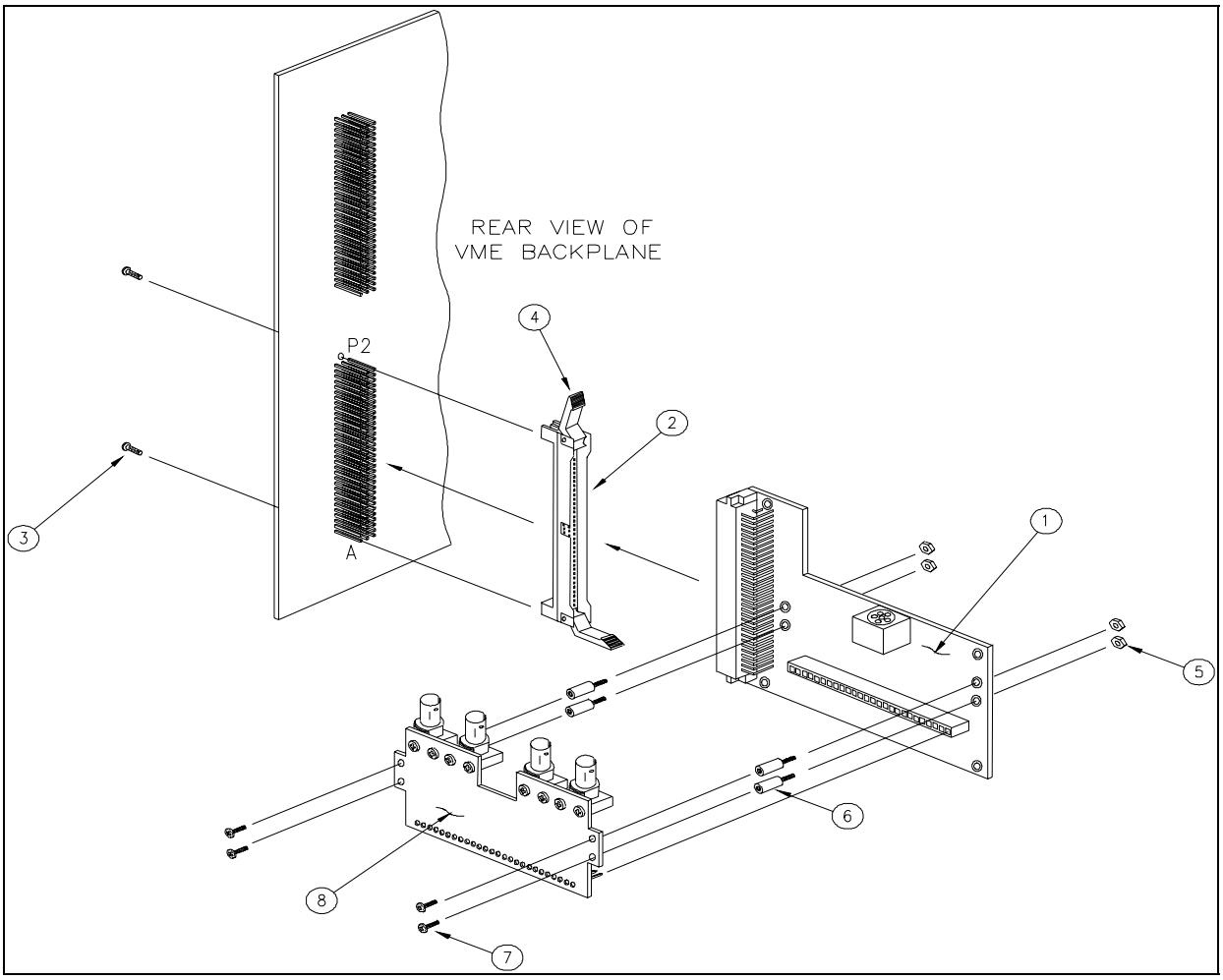


Figure D-3 Direct-attach P2 Cabinet Kit Installation

NOTE: The connector can go on two ways. When installed properly, the notch will be on the "P2 row" side of the connector. Correct installation will result in both the VME6U-board and cabinet-kit component sides facing the same direction. Improper orientation will result in damage. The cabinet-kit board must be installed on the P2 connector of the same slot as the host SCRAMNet board.

Additional details concerning features and installation are included in *SCRAMNet Network Cabinet Kit Hardware Reference*.

This page intentionally left blank

APPENDIX E

HOST TIMING ACCESS

TABLE OF CONTENTS

E.1 Introduction E-1
E.2 Dual-Port RAM Controller Module E-1
E.2.1 Contention E-1
E.3 Host Interface Logic to Shared Memory/CSR E-2
E.3.1 ASIC-Internal CSR READ E-3
E.3.2 ASIC-Internal CSR WRITE E-4
E.4 Host Interface Logic to Host Specific CSR E-5
E.4.1 Host-Specific CSR READ E-6
E.5 Access Times E-6
E.5.1 Typical Access Sequence E-7
E.5.2 Worst-Case Condition E-7
E.5.3 Back-to-Back Host READs E-9
E.5.4 Back-to-Back Host WRITEs E-11

FIGURES

Figure E-1 ASIC Resources E-2
Figure E-2 READ From Internal CSR E-3
Figure E-3 WRITE to Internal CSR E-4
Figure E-4 Non-ASIC Resources E-5
Figure E-5 READ From External CSR E-6
Figure E-6 Two Host WRITEs in Contention With Three Network WRITEs E-7
Figure E-7 Back-to-Back Host READs E-9
Figure E-8 Back-to-Back Host READs E-9
Figure E-9 Back-to-Back Host WRITEs E-11

TABLES

Table E-1 Dual Port RAM Controller Access Types E-1
Table E-2 Dual Port RAM Controller Maximum Access Times E-8

E.1 Introduction

The SCRAMNet+ host access timing is comprised of three separate module timings.

- Dual-Port RAM Controller (DPRC).
- Host Interface Logic to shared memory/CSR.
- Host Interface Logic to host-specific CSR.

The first module allows shared memory to be updated by the high-speed serial network without utilizing valuable CPU bus bandwidth. The second module is needed to interface shared memory and the ASIC internal CSRs to the host CPU bus. The third module is needed to interface the host-specific external CSRs to the host CPU.

E.2 Dual-Port RAM Controller Module

SCRAMNet+ memory is controlled by the DPRC. The DRPC has two ports: one for host access and one for network access. The DPRC arbitrates requests for these two ports on a first come, first serve basis. In case of a tie, the high-speed serial network has priority. The first port, which is enabled for READ/WRITE, is connected to the host CPU. The second port is WRITE ONLY and is connected to the high-speed serial network. There are three types of accesses to the DPRC. They are shown in Table E-1.

Table E-1 Dual Port RAM Controller Access Types

Access Type	Port	Cycle Time
Host WRITE	1	240 ns
Host READ	1	133 ns
Network WRITE	2	133 ns

E.2.1 Contention

The SCRAMNet+ Network, being an intelligent peripheral, does buffered WRITEs to shared memory. This speeds up host WRITEs by latching the data and address when a host WRITE is detected, and then replying to the host immediately without waiting for the DPRC to finish the WRITE activity. Since the WRITE buffering is only one level deep, it is possible that on back-to-back accesses the second access will be delayed until the buffer is available (i.e., when the current DPRC cycle is finished).

This phenomenon results in the stretching of two types of cycles outside the normal case:

BACK-TO-BACK WRITE CYCLE

In this cycle the second reply on the bus is held off until the first WRITE cycle has finished.

WRITE CYCLE FOLLOWED BY A READ CYCLE

In this case the READ cycle is delayed from starting until the previous WRITE cycle is completed.



NOTE: The cases resulting in stretched cycles describe a very fast host bus condition and are not normal. In reality, SCRAMNet+ memory was designed and optimized for CPU data storage. Therefore CPU activities such as instruction fetches, instruction execution, and other miscellaneous activities will ensure the phenomenon of cycle stretching will rarely, if ever, occur.

E.3 Host Interface Logic to Shared Memory/CSR

The second module is the actual host interface logic needed to interface the ASIC resources (memory and internal CSR) to the host CPU bus. This is the logic that translates all VME host transactions to SCRAMNet+. This host logic is responsible for determining the format of the access.

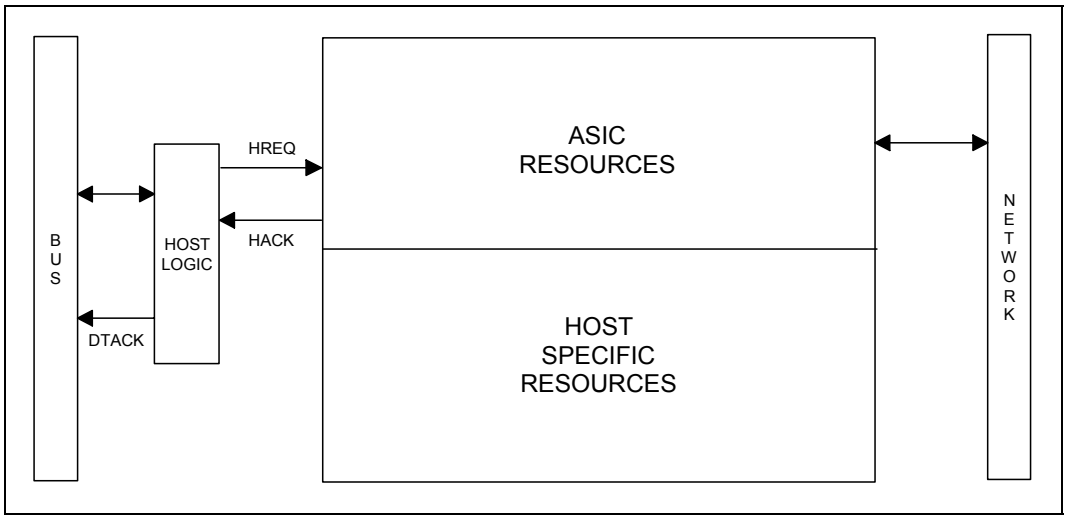


Figure E-1 ASIC Resources

This format is typically a READ or WRITE operation to the ASIC resources; the shared memory and CSRs.

E.3.1 ASIC-Internal CSR READ

The ASIC is typically configured through a number of control and status registers (CSR). The READ and WRITE cycles are not any different from any other ASIC resource cycles. The following is a READ cycle of an ASIC-internal CSR:

AS (Address Strobe) and DS (Data Strobe) fall
HREQ (Host Request) requested for a particular CSR (address)
Wait for HACK (Host Acknowledge) to provide READ data
Assert DTACK (Data Transfer Acknowledge)
Requester (Master/CPU) de-asserts AS and DS (end of cycle)

Figure E-2 is an example of a READ from an internal CSR.

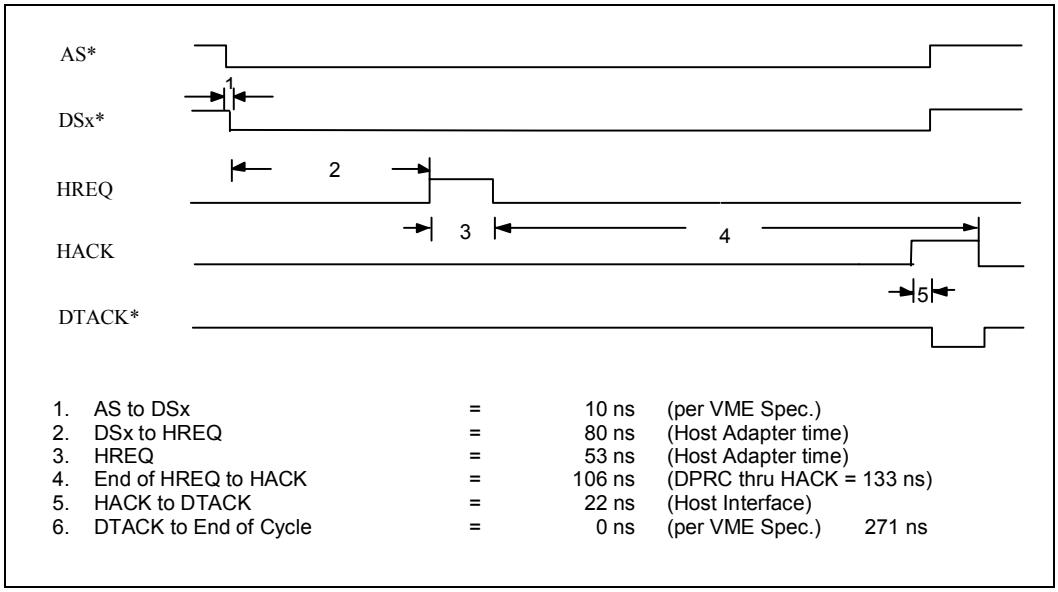


Figure E-2 READ From Internal CSR

E.3.2 ASIC-Internal CSR WRITE

This is a typical WRITE cycle of an ASIC-internal CSR:

AS and DS fall
HREQ requested for a particular CSR (address)
Data accepted and ASIC asserts HACK
Assert DTACK (Data Transfer Acknowledge)
Requester (Master/CPU) de-asserts AS and DS (end of cycle)

Figure E-3 is an example of a WRITE to an internal CSR.

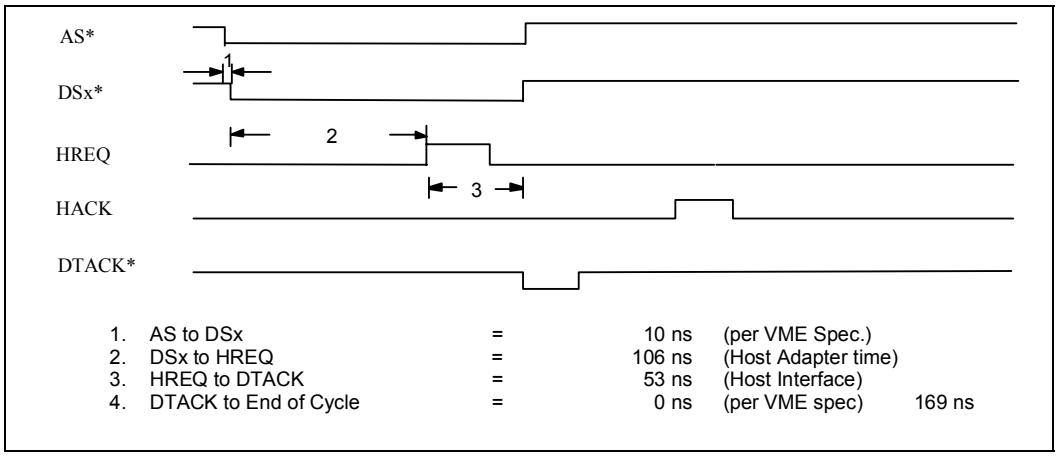


Figure E-3 WRITE to Internal CSR

E.4 Host Interface Logic to Host Specific CSR

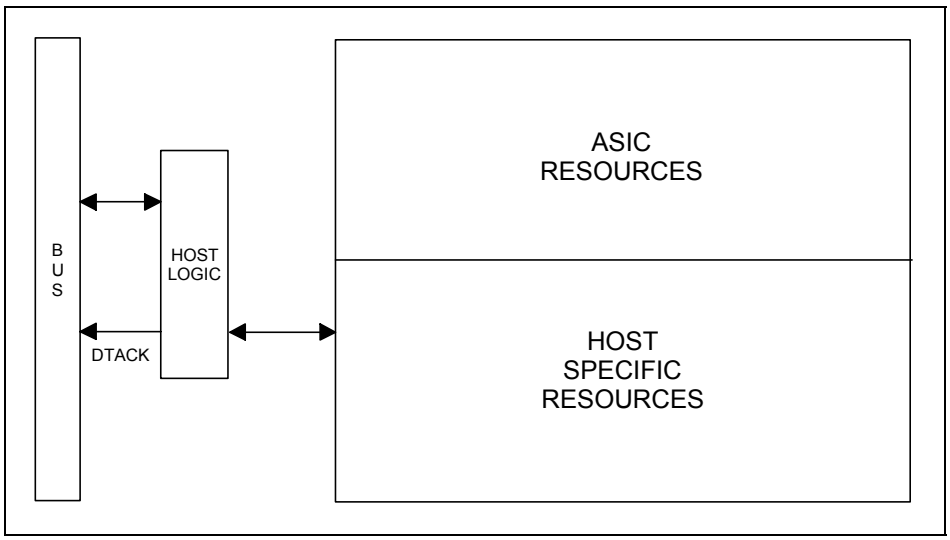


Figure E-4 Non-ASIC Resources

The third module is the actual host logic needed to interface the non-ASIC resources (external host-specific CSR) to the host CPU as depicted in Figure E-4. The Interrupt Vector register is a non-ASIC resource. Timings to this register vary significantly from that of ASIC-related resources.

E.4.1 Host-Specific CSR READ

READ operations on the external CSRs do not go through normal ASIC data paths. Therefore, timings for this cycle vary from other CSR READ cycles. These external CSRs are not ASIC resources and therefore do not request ASIC cycles to READ from them. Additional host logic monitors host access to these locations, and CSR data is provided directly to the host CPU without performing an ASIC Host Request (HREQ).

The following is the general format of an external CSR READ cycle:

AS and DS asserted
Additional host captures request
Provides READ data
Asserts DTACK
Requester (Master/CPU) de-asserts AS and DS

Figure E-5 is an example of a READ from an external CSR.

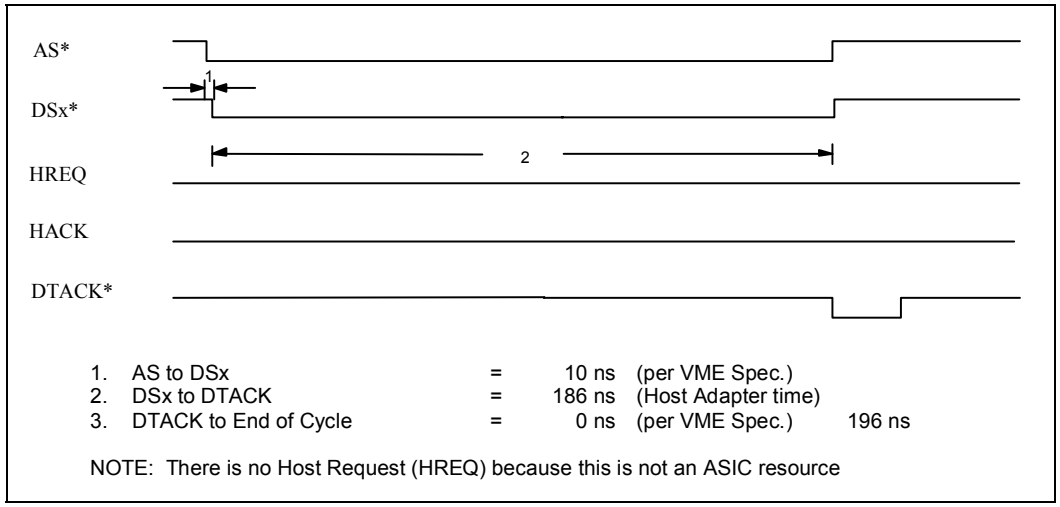


Figure E-5 READ From External CSR

E.5 Access Times

Host access to shared memory is affected by two things:

Priority on contention between the two ports
Asynchronous request handling

There are minimum and maximum times that depend on Network WRITE priority and the asynchronous handling of the requests. The minimum DPRC access times (i.e., no contention between ports) are shown in Table E-1. This table does not include host timing. The “worst case” scenario resulting in maximum access times would be a host WRITE followed immediately by another host WRITE, and simultaneous receipt of three network WRITES.

Minimum access times occur when there is no contention. Maximum access times result when the worst possible case of contention occurs.

The SCRAMNet+ cycle is initiated by the assertion of the DS and the de-assertion of the DS as the end of the cycle.

E.5.1 Typical Access Sequence

The following is the typical Host READ and Host WRITE sequence with no contention.

READ

AS and DS fall (Start of cycle)
HREQ asserted
HREQ_PEND (Host Request Pending) asserted (DPRC accepts request)
DPRC READ shared memory and pass the data to Host-Interface Logic
De-assert HREQ_PEND
HACK
Release bus (DTACK)
Requester/Host CPU de-assert DSx and AS (End of Cycle)

WRITE

AS and DS fall (Start of cycle)
HREQ asserted
DPRC buffer data and address
DPRC assert HREQ_PEND
HACK asserted
Release bus (DTACK)
De-assert HREQ_PEND
Requester/Host CPU de-assert DSx and AS (End of Cycle)
DPRC WRITE to shared memory
HACK

E.5.2 Worst-Case Condition

The scenario described in all these cases considers the "worst case" operating condition. This is a very unlikely condition, and would not occur in an average configuration. The maximum timing values were generated in a laboratory with a "worst case" scenario emulation. Figure E-6 is a functional diagram of the "worst case" scenario.

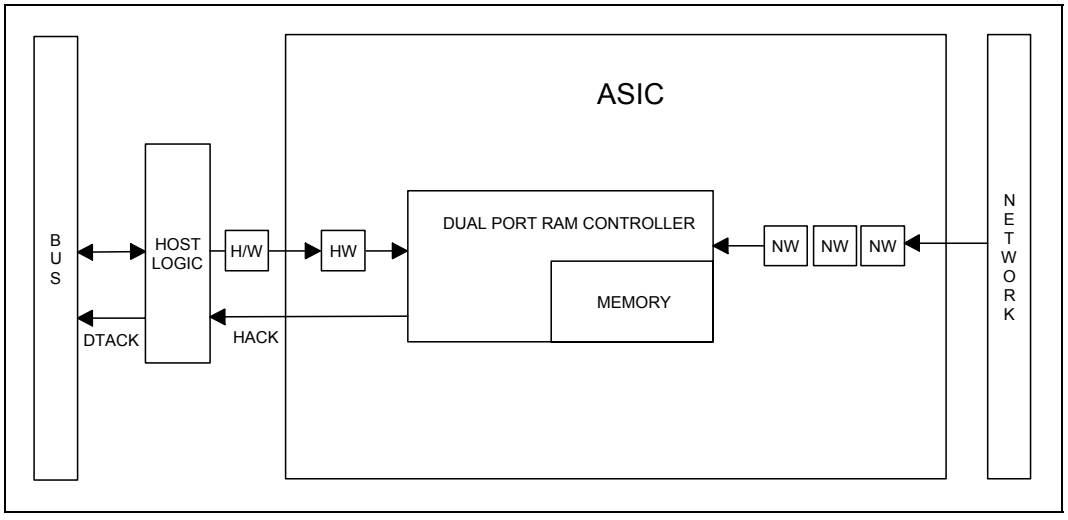


Figure E-6 Two Host WRITEs in Contention With Three Network WRITEs

CONFIGURATION

The test was conducted with two nodes on a ring and an extremely fast host CPU as a host data generator. This fast CPU is theoretical since there is no CPU available today that is capable of maintaining the VME specification of 40 MB/sec. The two nodes were defined having their transmitters disabled, and their Interrupt FIFOs filled. The nodes were configured to transmit in Plus mode, which translates to the SCRAMNet+ maximum throughput of 16.6 MB/sec. A Plus mode packet has new data every 240 ns. The "worst case" condition, which is **not a standard configuration**, consists of releasing the FIFOs and having the ideal-host CPUs performing back-to-back host read and write operations.

GENERATED EFFECTS

The effect of the above setup will result in:

Back-to-Back Host READs. In this case, the "worst case" timing would occur where two Network WRITE operations will stretch the second host READ operation. The third and fourth host READ operations will toggle between network WRITE operations. This pattern will repeat itself, if the "worst case" scenario continues, every fifth host READ.

(i.e., HR1, NW1, NW2, HR2, NW3, HR3, NW4, NW5, HR4...)

Back-to-Back Host WRITEs. In this case, the "worst case" timing would be where three Network WRITE operations will stretch the second host WRITE operation. The third and fourth host WRITE are also held off by two Network WRITE operations each. This pattern will repeat itself every fifth host WRITE if the "worst case" scenario continues.

(i.e., HW1, NW1, NW2, NW3, HW2, NW4, NW5, HW3, NW6, NW7, HW4...)

The maximum time for a host READ or WRITE would occur with host back-to-back HREQs within 30 ns, combined with multiple Network WRITEs. Port two would be given priority so that the network would not be delayed by host activity to the shared memory.

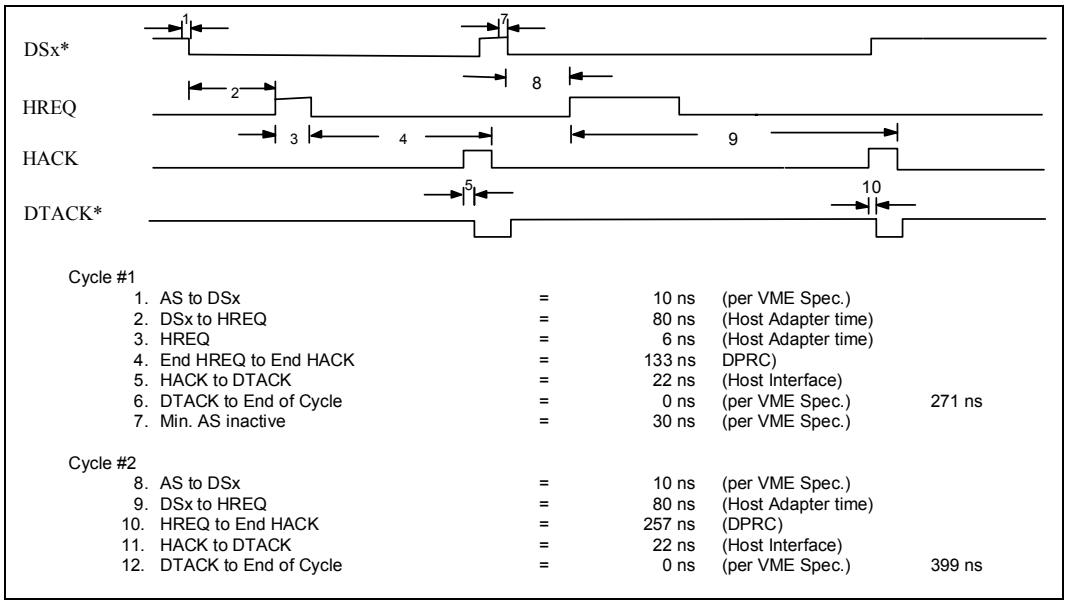
Maximum access times for the DPRC based on the "worst case" scenario are shown in Table E-2 (These calculations do not include host timing):

Table E-2 Dual Port RAM Controller Maximum Access Times

Access Type	Cycle Time	Access Type	Cycle Time
Host READ	133 ns	Host WRITE	240 ns
2 Network WRITEs	+266 ns	3 Network WRITEs	+399 ns
+ Host READ	+133 ns	Host WRITE	+240 ns
Maximum Time	532 ns	Maximum Time	879 ns

E.5.3 Back-to-Back Host READs

Figure E-7 shows access timing for a Back-to-back VME read. All timing shown is based on the fastest possible VME master (according to VME specification C.1) driving the bus, and the maximum bandwidth on the SCRAMNet+ Network ring.



Cycle #1

1.	AS to DSx	=	10 ns	(per VME Spec.)	
2.	DSx to HREQ	=	80 ns	(Host Adapter time)	
3.	HREQ	=	6 ns	(Host Adapter time)	
4.	End HREQ to End HACK	=	133 ns	DPRC)	
5.	HACK to DTACK	=	22 ns	(Host Interface)	
6.	DTACK to End of Cycle	=	0 ns	(per VME Spec.)	271 ns
7.	Min. AS inactive	=	30 ns	(per VME Spec.)	

Cycle #2

8.	AS to DSx	=	10 ns	(per VME Spec.)	
9.	DSx to HREQ	=	80 ns	(Host Adapter time)	
10.	HREQ to End HACK	=	257 ns	(DPRC)	
11.	HACK to DTACK	=	22 ns	(Host Interface)	
12.	DTACK to End of Cycle	=	0 ns	(per VME Spec.)	399 ns

Figure E-7 Back-to-Back Host READs

Figure E-8 illustrates how the DPRC must manage two inputs. The network WRITEs have the higher priority. Consequently, in this "worst case" scenario, the Host READ cycle is stretched.

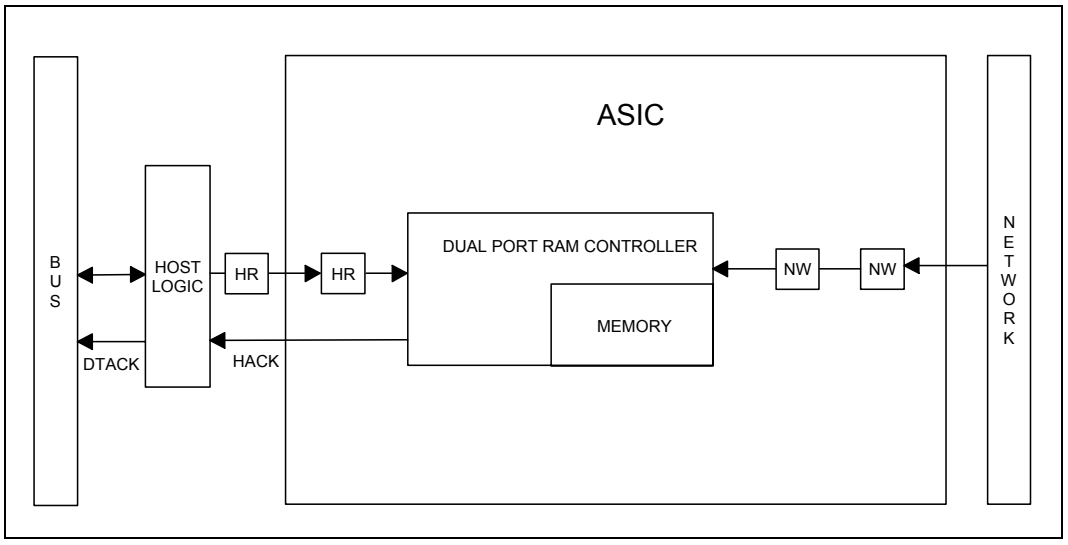


Figure E-8 Back-to-Back Host READs

FIRST CYCLE

1. AS, DSx fall – Start of Cycle (80 ns delay to decode address).
2. HREQ asserted – READ request to DPRC.
3. Host interface logic sleeps until HACK is received from DPRC (Maximum READ Time)
4. DPRC asserts HREQ_PEND Request accepted
   - DPRC is backed up with two Network WRITEs
   - DPRC performs READ for host.
5. HACK is issued to host interface logic.
6. DTACK is asserted – bus released.
7. DTACK is de-asserted; AS, DSx de-asserted – End of Cycle.

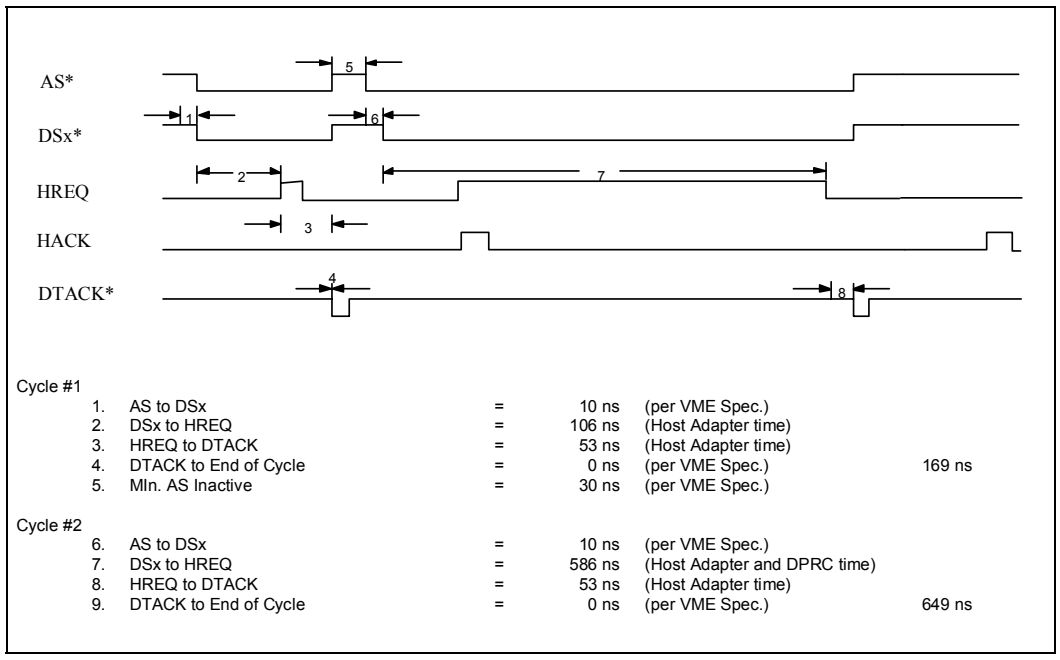
SECOND CYCLE

8. AS, DSx fall - Start of Cycle (80 ns delay to decode address).
9. HREQ asserted - READ request to DPRC.
10. Host interface logic sleeps until HACK is received from DPRC (Maximum READ Time) DPRC is processing two Network WRITEs.
11. HACK is issued to host interface logic; DTACK is de-asserted – bus released.
12. DTACK is asserted; AS, DSx de-asserted - End of Cycle.

E.5.4 Back-to-Back Host WRITEs

Figure E-9 shows access timing for two back-to-back VME WRITEs as fast as possible according to the VME specification revision C1. The first cycle shows the normal buffered WRITE cycle and the second cycle shows a stretched WRITE cycle (maximum hold). As in the VME READ, the host interface logic waits for the first DS to fall and then delays 106 ns to allow address decode and attain bus stability.

In the first VME cycle, the host logic raises a host WRITE request to the DPRC. The DPRC accepts the request, buffers the data, and issues a HACK to the Host Interface logic. The host interface logic responds by asserting DTACK 53 ns later. Under normal conditions this saves CPU cycles. By VME specification, as soon as the VME bus master detects DTACK on the bus, the AS and DSx may then be re-asserted to end the cycle.



Cycle #1					
1.	AS to DSx	=	10 ns	(per VME Spec.)	
2.	DSx to HREQ	=	106 ns	(Host Adapter time)	
3.	HREQ to DTACK	=	53 ns	(Host Adapter time)	
4.	DTACK to End of Cycle	=	0 ns	(per VME Spec.)	169 ns
5.	MIn. AS Inactive	=	30 ns	(per VME Spec.)	
Cycle #2					
6.	AS to DSx	=	10 ns	(per VME Spec.)	
7.	DSx to HREQ	=	586 ns	(Host Adapter and DPRC time)	
8.	HREQ to DTACK	=	53 ns	(Host Adapter time)	
9.	DTACK to End of Cycle	=	0 ns	(per VME Spec.)	649 ns

Figure E-9 Back-to-Back Host WRITEs

FIRST CYCLE

1. AS, DSx fall - Start of Cycle (106 ns delay to decode address).
2. HREQ asserted - WRITE request to DPRC
   - DPRC asserts HREQ_PEND Request accepted
   - DPRC buffers data
   - DPRC asserts HACK.
3. Host interface logic generates DTACK 53 ns later.
4. AS and DSx are de-asserted.
5. Minimum time AS is inactive is 30 ns.

SECOND CYCLE

The second cycle shows a second host WRITE is initiated 30 ns later according to minimum VME requirements. Under normal operating conditions it is unlikely the host will be able to turn around that quickly. An HREQ cannot be generated since the first one is still being serviced. Additionally, in this “worst case” scenario, three network transactions must be processed before the previous host request is cleared.

Once the network transactions are cleared, the buffered data from the host WRITE can be written to shared memory which clears the WRITE buffer. This takes 240 ns from the request to the acknowledgment.

DPRC has not yet written to shared memory

6. AS, DSx fall - Start of second host cycle.
7. HREQ cannot be asserted since the previous request is still pending.
   - DPRC receives 3 network WRITEs
   - DPRC processes network WRITEs
   - DPRC performs host WRITE to shared memory
8. HREQ_PEND de-asserted (No longer busy)
   - DPRC asserts HREQ_PEND - WRITE request accepted
   - DPRC buffers data
   - DPRC asserts HACK
   - Host Interface logic generates DTACK
9. AS and DSx are de-asserted

APPENDIX F

CONFIGURATION AIDS

SCRAMNet+ CONTROL/STATUS REGISTERS REFERENCE SHEET

CSR 0

0	RX ENB
1	TX ENB
2	REDUND LINK TOGGLE
3	HOST INT ENB
4	AUX CTRL RAM ENB
5	INT MEM MASK MATCH
6	OVRD RIE FLAG
7	INT ON ERRORS
8	NET INT ENB
9	OVRD TIE FLAG
10	ENB TX DATA FILTER
11	ENB LOWER 4K FILTER
12	RST TX/RX FIFO
13	RST INT FIFO
14	RST TX FIFO
15	INSERT NODE

CSR 2

0	available to host
1	available to host
2	available to host
3	available to host
4	available to host
5	available to host
6	DSB FO LPBCK
7	ENB WIRE LPBCK
8	DSB HOST TO SM WRT
9	ENB WRT OWN SLOT
10	ENB INT RX OWN SLOT
11	MSG LENGTH LIMIT
12	VAR LENGTH MSGS
13	ENB HIPRO WRITE
14	MULT NATIVE MSGS
15	NO NTWK ERR CRCT

CSR 4

0	always 0
1	always 0
2	RFA 2
3	
4	
5	RX FIFO ADDRESS FIELD
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	RFA 15

CSR 6

0	data intrpt vector
1	data intrpt vector
2	data intrpt vector
3	data intrpt vector
4	data intrpt vector
5	data intrpt vector
6	data intrpt vector
7	data intrpt vector
8	reserved
9	reserved
10	reserved
11	reserved
12	reserved
13	reserved
14	reserved
15	reserved

ACR

0	RIE
1	TIE
2	EXT TRG 1
3	EXT TRG 2
4	HIPRO ENB
5	reserved
6	reserved
7	reserved

LED STATUS

G	INSERT
G	MESSAGE WAITING
G	CARRIER DETECT
Y	ERROR
G	NATIVE MESSAGE
G	FOREIGN MESSAGE

CSR 1 (READ RESET)

0	TX FIFO FULL
1	TX FIFO NOT EMPTY
2	TX FIFO 7/8 FULL
3	always 0
4	INT FIFO FULL
5	PROTOCOL VIOLATION
6	CARRIER DETECT FAIL
7	BAD MESSAGE
8	RX OVERFLOW
9	TX RETRY
10	TX RETRY TIME-OUT
11	REDUND TXRX FAULT
12	GP CTR/TIMER OVRFLO
13	CURRENT LINK FOR USE
14	INTERRUPTS ARMED*
15	FO BYPASS NOT CNCTD

CSR 3

0	NN0
1	
2	NUMBER
3	OF
4	NODES
5	
6	
7	NN7
8	TXID0
9	
10	
11	NODE ID
12	
13	
14	
15	TXID7

CSR 5

0	RFA16
1	
2	RX FIFO
3	ADDRESS
4	(MSW)
5	
6	RFA22
7	reserved
8	reserved
9	reserved
10	reserved
11	reserved
12	reserved
13	reserved
14	RF RETRY
15	INT FIFO NOT EMT

CSR 7

0	error intrpt vector
1	error intrpt vector
2	error intrpt vector
3	error intrpt vector
4	error intrpt vector
5	error intrpt vector
6	error intrpt vector
7	error intrpt vector
8	reserved
9	reserved
10	reserved
11	reserved
12	reserved
13	reserved
14	reserved
15	reserved

* Write to CSR1 to re-arm interrupts.

CSR 8

0	AGE & RXID MUX
1	HOLDOFF DISABLE
2	CHP SELECT EEPROM
3	AUX MICROWIRE
4	MICROWIRE DOUT
5	EEPROM PROG ENABLE
6	MICROWIRE CLOCK LN
7	MICROWIRE DOUT DIN
8	INIT ASIC/CSR RESET
9	GP CTR FREE
10	RX INT OVERRIDE
11	MECH SW OVR
12	MEM SIZE
13	MEM SIZE
14	MEM SIZE
15	Reserved

CSR 10

0	SM ACCESS ENB
1	reserved
2	reserved
3	reserved
4	reserved
5	reserved
6	reserved
7	reserved
8	reserved
9	reserved
10	reserved
11	reserved
12	SMA 12
13	SM ADDRESS
14	(LSP)
15	SMA 15

CSR 12

0	VIRT PG ENB
1	always 0
2	always 0
3	always 0
4	always 0
5	VPA 12
6	
7	
8	VIRTUAL
9	PAGE
10	NUMBER
11	
12	
13	
14	
15	VPA 22

CSR 14

0	reserved
	reserved
	reserved
	reserved
	reserved
	reserved
	reserved
	reserved
	reserved
	reserved
	reserved
	reserved
	reserved
	reserved
	reserved
15	reserved

CSR 9

0	TX FIFO FULL MASK
1	TX FIFO NOT EMP MASK
2	TX FIFO 7/8 FULL MASK
3	BIST STREAM (R/O)
4	RX FIFO FULL MASK
5	PROTOCOL VIOL MASK
6	CARRIER DETECT FAIL MASK
7	BAD MESSAGE MASK
8	RX OVERFLOW MASK
9	TX RETRY MASK
10	TX RETRY TIME-OUT
11	REDUN TXRX FAULT MASK
12	GP CTR/TIMER OVRFLO
13	UTIL CTR MODES
14	UTIL CTR MODES
15	FO BYPASS NOT CNCTD MASK

CSR 11

0	SMA 16
1	
2	
3	
4	
5	
6	SHARED
7	MEMORY
8	ADDR
9	(MSP)
10	
11	
12	
13	
14	
15	SMA 31

CSR 13

0	RD COUNT 0
1	
2	
3	
4	
5	GENERAL
6	PURPOSE
7	COUNTER/
8	TIMER
9	REGISTER
10	
11	
12	
13	
14	
15	RD COUNT 15

CSR 15

0	not used
1	VME
2	IRQ
3	REGISTER
4	Levels
5	1-7
6	only
7	
8	reserved
9	reserved
10	reserved
11	reserved
12	reserved
13	reserved
14	reserved
15	reserved

CSR 16

0	HIPRO READ ENB
1	HIPRO READ ACR ENB
2	reserved
3	reserved
4	reserved
5	reserved
6	reserved
7	reserved
8	reserved
9	reserved
10	reserved
11	reserved
12	reserved
13	reserved
14	reserved
15	reserved

SCRAMNet+ NETWORK CONFIGURATION DATA SHEET

NODE ID	HOST MACHINE	MEMORY ADDRESS & BUS	MEMORY SIZE	CSR ADDRESS & BUS	INT LEVEL	SCRAMNet SERIAL #

GLOSSARY

A16 ----------A type of module that provides or decodes an address on address lines A01 through A15.

A24 ----------A type of module that provides or decodes an address on address lines A01 through A23.

A32 ----------A type of module that provides or decodes and address on address lines A01 through A31.

ACR ----------Auxiliary Control RAM. A memory buffer typically used as a data bus width extension for control purposes only. Also referred to as shadow memory.

backplane ----------A printed circuit board (pcb) with 96-pin connectors and signal paths that bus the connector pins. Some systems have a single pcb, called the J1 backplane. It provides the signal paths needed for basic operations. Other systems also have a second pcb, called a J2 backplane. It provides the additional 96-pin connectors and signal paths needed for wider data and address transfers. Still others have a single pcb, called a J1/J2 backplane that provides the signal conductors and connectors of both the J1 and J2 backplanes.

bad message ----------A message error condition reported by a node's receiver circuitry. This condition is automatically corrected by SCRAMNet+ hardware.

board----------A pcb, its collection of electronic components, and either one or two 96-pin connectors that can be plugged into the backplane connectors.

burst----------A protocol where messages are transmitted without error correction to gain higher throughput.

burst+----------Also burst plus. A variable packet size enhancement for the burst protocol. Maximum packet size may be set to either 256 bytes or 1024 bytes.

carrier wave ----------An electromagnetic wave that can be modulated, as in frequency, amplitude, or phase, to transmit data, images, sound, or other signals.

CSR ----------Control/Status Register.

CTB ----------Control Transfer Bus.

data filter----------A process of comparing a host WRITE to shared memory with contents of the specified memory location to eliminate transmission of redundant data and reduce network traffic.

deterministic ----------Completely predictable message transit time from application to application.

device interrupt----------An interrupt received on interrupt priority levels 20-23. Device interrupts can be requested only by devices, controllers, and memories.

DPRC ----------Dual Port RAM Controller.

DS----------Data Strobe.

DTACK----------Data Transfer Acknowledge.

DTB ----------Data Transfer Bus.

FIFO ----------A data storage method; First In First Out. Also refers to the specific storage area; Transmit FIFO, Interrupt FIFO, etc.

foreign message----------A message that is in (passing through) a node other than the one of origin.

HACK ----------Host Acknowledge.

HREQ----------Host Request.

HREQ_PEND----------Host Request Pending.

insert a node----------The act of placing a node on a network for the purpose of transmitting and receiving messages.

interrupt ----------An event that changes the normal flow of instruction execution other than an exception or a branch, jump, case or call instruction.

interrupter----------A functional module that generates an interrupt request on the priority interrupt bus and then provides states/ID information when the interrupt handler requests it.

ISR ----------Interrupt Service Routine. A routine executed when a device interrupt occurs.

latched----------Data is electrically stored in a circuit until it is needed. A method of coordinating two synchronous events.

longword ----------Four bytes (32 bits) of data.

loopback----------A method of transmitting to the same node's receivers for testing purposes. Applies to both fiber optic and wire media. Also, a test that loops the outgoing signal back to its source.

LSB----------Least Significant Byte.

LSP----------Least Significant Part of the interrupt address.

LSW----------Lease Significant Word.

master----------A functional module that initiates DTB cycles to transfer data between itself and a slave module.

message packet ----------See packet.

MSB----------Most Significant Byte.

MSP ----------Most Significant Part of the interrupt address.

MSW----------Most Significant Word.

native message ----------A message that is received by the node of origin.

node latency ----------The time delay at a node before a foreign message can be retransmitted.

packet----------A message that travels on the network. The minimum packet consists of 81 bits and 1 start bit. The packet includes five fields: Source ID (8 bits), Age (8 bits), Control (3 bits), Data Address (21 bits), Data (32 bits), and 9 parity bits; one for every 8 bits.

pcb----------Printed circuit board.

physical address ----------The address used by hardware to identify a location in physical memory or on directly-addressable secondary storage devices (such as disks). A physical memory address consists of a page-frame number and the number of a byte within the page.

platinum----------A protocol where messages are transmitted as fast as the system will allow with error correction enabled.

platinum+ ----------(Also platinum plus). A variable packet size enhancement for the platinum protocol. Maximum packet size may be set to either 256 bytes or 1024 bytes.

protocol violation----------A signal error at the physical layer (fiber or coax) resulting from noise on the transmission lines or a result of hardware failure. This violation can be any one of the following:

- Missing transition for two clock periods on either line
- Parity error
- Framing error

read cycle ----------A DTB cycle used to transfer 1-, 2-, 3-, or 4-bytes from a slave to a master. The cycle begins when the master broadcasts an address and an address modifier. Each slave captures this address and address modifier, and checks to see if it is to respond to the cycle. If so, it retrieves the data from its internal storage, places it on the data bus, and acknowledges the transfer. Then the master terminates the cycle.

requester ----------A functional module that resides on the same board as a master or interrupt handler and requests use of the CTB whenever its master or interrupt handler needs it.

retry ----------A hardware failure condition reported when the first attempt to send a message around the network has resulted in some type of bit error. The message will be retransmitted indefinitely by the originating node until it is received correctly by the originating node. Valid only in error correction mode (PLATINUM.)

retry time-out ----------A hardware failure condition reported when the first attempt to send a message around the network is not received by the originating node within the time out period specified in CSR5. The message will be retransmitted indefinitely by the originating node until it is received correctly by the originating node. Valid only in error correction mode (PLATINUM.)

ring time----------The time it takes a message to traverse the network ring from the originating node and back again. The time can be calculated by estimating propagation delay at 5 ns per meter of cable plus 250 ns per node. This assumes a best case using 32-bit standard data with no other nodes transmitting. Worst case would be using 800 ns per node. Other times can be calculated using the maximum delays per node when sending variable length data.

rising edge ----------The time during which a signal makes its transition from low to high.

Rx----------Abbreviation for receive or receiver.

shadow memory ----------See Auxiliary Control RAM (ACR)

shared memory (SM)----------SCRAMNet memory physically located on the network board. This dual-ported memory is accessible by the host and the network. A host WRITE to shared memory results in a transmitted WRITE to all SCRAMNet nodes at the same relative location.

shortword ----------16 bits. Also referred to as halfword.

signal mnemonics ----------Terms used to identify signal line events. (1) An asterisk following the name of signals that are level-significant denotes the signal is true/valid when the signal is low. (2) An asterisk following the name of signals that are edge-significant denotes the actions initiated by that signal occur on the falling edge.

slave ----------A functional module that detects DTB cycles initiated by a master and, when those cycles specify its participation, transfers data between itself and the master.

slot ----------A position where a board can be inserted into a backplane. If the system has both a J1 and a J2 backplane (or a combination J1/J2 backplane) each slot provides a pair of 96-pin connectors. If the system has only a J1 backplane, then each slot provides a single 96-pin connector. Also, another name for message packet.

time-out----------Also network time-out. The time written to CSR5 that must elapse before a native message will be retransmitted. The time-out must be a non-zero value.

Tx ----------Abbreviation for transmit or transmitter.

UAT ----------A master that sends or receives data in an unaligned fashion.

VME address space----------The VME address space varies according to specific VME device and is identified as A16, A24, or A32 space. A32 is the largest address space; it allows up to 4 gigabytes of space using 32 bit addresses. A24 space uses 24 bit addresses, and A16 space uses 16 bit addresses.

VMEbus----------A standard bus by which small computers and intelligent peripheral devices can be connected. The term VME stands for Versa Module Eurocard. This non-proprietary bus conforms to the American National IEEE Standard 1014 (ANSI/IEEE std 1014).

write cycle----------A DTB cycle used to transfer 1-, 2-, 3-, or 4-bytes from a master to a slave. The cycle begins when the master broadcasts an address and address modifier and places data on the DTB. Each slave captures this address and address modifier, and checks to see if it is to respond to the cycle. If so, it stores the data and then acknowledges the transfer. The master then terminates the cycle.

INDEX

A

access timing E-1, E-9, E-11
ACR..... 1-1, 2-4, 2-5, 2-6, 2-8, 2-10, 4-8, 5-7, 5-8, 5-9, 5-10, 5-12, 5-15, 5-16, 5-19, B-3, B-8, B-17, C-11, F-1, F-2
  enable B-3, C-1
  location 2-10, 5-19
active status B-6
address
  boundary 2-3, 3-3, 5-19, B-3
  space 2-1, 3-3, 4-9, 4-14, 5-17
  strobe A-2, E-3, E-4, E-6, E-7, E-10, E-11, E-12
ASIC
  resources E-2
  internal CSR E-3, E-4
auxiliary
  connection 4-4, 4-12
  connector 3-5, 5-15

B

backplane 3-3, 4-9, 4-13, 4-15
  connectors 3-3
bad message 5-14, B-3, B-5, B-12, C-2, C-8
bandwidth 2-4, 5-5, 5-19, A-1, E-1, E-9
battery power 4-4
bit errors B-5
buffered transmit messages B-2
built-in self test B-12, C-8, F-2
BURST mode 2-4, 3-1, 4-13, 5-4, 5-5, 5-6, 5-7, 5-10, B-8
BURST PLUS mode 2-4, 3-1, 4-13, 5-7, B-8, C-3
bus resolution switch 4-1

C

cabinet kit 1-1, 3-5, 4-4, 4-9, 4-10, D-1, D-2, D-3
  board 4-4, D-3
  direct attached D-3
  expanded D-2
  options D-1
cable connections
  fiber-optic 5-3
cables
  fiber-optic 2-9, 2-11, 4-2, 4-10, 5-25
carrier detect B-5, C-2
carrier detect fail 5-4, 5-10, 5-14, B-12, C-8
coaxial cable 4-10, 5-4
connector
  20-pin D-1
  DIN 41612 ..... D-3
  J302 ..... A-5
  J8 A-5
  P2A-5, D-3
  receiver input 2-11, 5-25
  SMA 4-10
  transmitter output 2-11, 5-25
control registers 5-3, 5-6
control/status register . 2-1, 3-1, 3-3, 4-5, 4-13, 5-3, B-1
count errors B-12, C-8
count triggers C-8
counter mode B-11
counter register 5-17
counter/timer 5-16, 5-17
CSR addressing 4-5

D

data
  filter 2-3, 2-9, 3-1, 5-12, 5-17, 5-18, B-4, C-1
  size 2-4, 5-5
  strobe E-3, E-4, E-6, E-7, E-10, E-11, E-12
  transfer ACK . E-3, E-4, E-6, E-7, E-10, E-11, E-12
  transfers 3-3
  variable size 5-3
decoder 4-14
device driver 2-5
down-stream node 4-11
dual fiber-optic cables 2-4
dual port
  memory controller 2-1
  memory 3-2
  RAM controller E-1, E-6, E-7, E-8, E-9, E-10, E-11, E-12
dual-vector memory 3-2

E

EEPROM
  initialization program 4-13
  read 4-1, 4-4
  write 4-1, 4-4
electrical requirements A-1
electronic bypass switch 3-4
enable interrupt on own slot 2-6, 5-9
error
  conditions 2-7, 2-9, 5-3, 5-4, 5-10, 5-14, B-5
  correction 5-6, A-1, B-8, C-3
  interrupt mask 3-2
external trigger 2-5, 2-6, 2-8, 4-8, 5-8, 5-12, 5-15
  connections 4-1
  trigger 1 ..... B-3, C-11
  trigger 2 ..... B-3, C-11

F

faceplate D-1, D-2
fiber optic
  bypass switch 2-11, 3-1, 4-1, 4-2, 4-11, 4-12, 5-4, 5-14, 5-25, A-8, B-6, B-7, B-12, C-2, C-8
  cable 4-10, 4-11, 5-3, 5-4
  cabling 2-9, 5-4, 5-10
  loopback mode 5-20, 5-24, B-6

FIFO
buffers 2-3, 5-7
interrupt 2-3, 5-9, 5-12, 5-14, 5-15, B-2, C-6
pointer 5-15
receive 2-3, B-5
receive/transmit....B-4
transceiver....2-3, 5-7
transmit.. 2-3, 2-8, 2-10, 4-14, 5-6, 5-7, 5-14, 5-26, 5-28, B-2, B-4, B-5, B-9, B-12, C-1, C-2, C-6, C-8
fixed-length message....2-4, 3-1, 5-5, A-1
forced interrupt method 2-8, 5-12
foreign message 2-4, 2-7, 5-6
framing error B-5

G

general purpose counter/timer....2-8, 3-2, 5-14, 5-16, 5-17, B-1, B-6, B-11, B-12, B-15, C-2, C-7, C-8

H

handshaking logic 2-1
high-speed transfer....2-1
HIPRO mode....2-3, 2-9, 2-10, 4-13, 5-8, 5-15, 5-19
host
acknowledge E-3, E-4, E-7, E-10, E-11, E-12
adapter board D-3
computer....2-1, 4-9, 5-3, 5-14
cycles....2-10, 5-19
interface board 4-4, D-1
interface logic E-1, E-2, E-5
interrupt enable B-2, C-1
memory....2-9, 5-28
power....4-4
processor.... 1-1, 2-1, 2-5, 2-9, 5-10, 5-12, 5-19
request pending E-7
request E-3, E-4, E-6, E-7, E-10, E-11, E-12
timing E-6, E-8
humidity range....A-1

I

I/O control 3-3
incoming interrupt....2-7
input/output 2-1
insert mode....5-20, 5-25, 5-26
insert node....2-11, 5-25, B-4, B-7, C-1
install 1-1, 4-6, 4-7
internal clock speeds A-1
interrupt
action....2-5, 5-8
address....2-3, 5-3
LSP....B-1, B-9, C-5, F-2
MSP....B-1, B-9, C-6, F-2
bit2-3, 2-6, 2-8, 5-10, 5-12
control 2-4, 5-9
enable 5-15
FIFO2-3, 2-6, 4-14, 5-7, 5-9, 5-12, 5-14, 5-15, B-2, B-3, B-4, B-5, B-9, B-12, C-1, C-2, C-5, C-6, E-8
handling 2-1
locations....2-8, 5-3, 5-12
message....2-5, 2-6, 5-9, 5-12
messages 2-8, 5-9
method
forced 2-8, 5-12
selected....5-12
on errors 3-3, C-1
operation 5-8, 5-9, B-6
options....4-13
request level B-10
request....B-1, B-2, B-3, B-10, B-16, C-11, F-2
results....5-8
service routine 2-5, 2-8, 5-1, 5-14, 5-15, 5-29
servicing....5-3
signal....5-10
vectors.... 2-5, 5-3, 5-10, 5-14, B-1, B-10, C-6, E-5

J

jumper 1-1, 4-1, 4-7
ground....4-1
J5 .... A-6
memory configuration....4-1
software compatibility....4-1
variable length message 4-1

L

latched B-5, B-6
latched error conditions....5-10
latching....E-1
least significant bit.... B-1
LED
carrier detect....2-9
error 2-9
foreign message....2-9
indicators.... D-1
message waiting....2-8
native message....2-9
light signal 4-9
long-link fiber 3-4, 4-10
longword....B-8, B-17, C-11
longword boundary....2-10, 5-5, 5-19
loopback mode2-10, 2-11, 5-4, 5-10, 5-20, 5-25, B-2, B-7

M

maximum node separation....4-10, A-1
maximum packet size....2-4
mechanical switch ... 2-10, 2-11, 5-20, 5-23, 5-24, 5-25
loopback mode 5-20, 5-23, B-11, C-7
media card 2-10, 2-11, 3-4, 4-1, 4-2, 4-4, 4-9, 4-12, 5-23, A-1, A-2, A-5, D-1, D-3

memory
address 2-1, 2-3, 3-3, 4-14, 5-1, 5-3, 5-5, 5-15, 5-16, B-11
base address 4-1
common blocks 5-3
expansion 2-1
on-board 2-1
page size 5-3
pages 5-3
size 2-1, B-11, B-14
update B-1, B-10, C-6
upgrade 3-1, 3-2, 4-2
usage 5-3
message
format 5-4
length A-1, B-7
packet 2-4, 3-5, 5-4, B-5
slot 2-4
microwire C-7, F-2
mode control 5-3
mode operation 5-3, 5-6
monitor and bypass mode 5-20
multiple nodes 2-4, 5-7
multiple packets 5-6

N

native message 2-3, 2-4, 5-12
network
access E-1
communications mode B-2
error correction C-3
errors 5-14
events B-12, C-8
interrupt enable B-3, C-1
message 2-8, 2-10, 5-2, 5-9, 5-12, 5-16, 5-19, B-3, B-11
throughput 2-4, 2-10, 5-17
time-out 4-1, 4-13, 4-14
traffic 2-8, 2-9, 5-17, B-2
transaction E-11, E-12
node ID 5-4, B-9, C-4, F-1
node identification 4-1, 4-14
node latency 5-5, 5-6, 5-7, 5-17, A-1
maximum 5-6
node memory 2-9, 2-11
non-ASIC resources E-5, E-6

O

operate 1-1, 5-1
originating node 2-5, 2-11, 5-4, 5-6, 5-28, B-5, B-7
oscilloscope 5-16
outgoing interrupt 2-6, 2-7
override receive interrupt enable 2-8
override transmit interrupt enable 2-8

P

P2 backplane connection D-2
parity error B-5
phantom bits 2-5
physical dimensions A-1
physical layer B-5
pinout A-5, A-6, A-7
PLATINUM mode 2-4, 3-1, 5-5, 5-6, 5-7, A-1, B-8, C-3
PLATINUM PLUS mode 2-4, B-8
PLUS mode 2-4, 5-5, 5-6
PLUS mode protocol 2-4, 5-5
power
battery 4-4
host 4-4
power options 4-4
power-up B-4
program size 5-3
propagation delay 5-7
protocol
register-insertion 2-4
protocol on/off 4-13
protocol violation ..... 5-14, B-3, B-5, B-12, C-2, C-

Q

quad switch 3-4, 3-5, 5-4, 5-27, 5-28

R

real-time applications 2-4, 3-1
real-time computer 2-1
receive
enable C-1
interrupt logic 5-12
interrupt override 2-8, 5-9
interrupt 2-5, 2-6, 5-9, B-3, B-11, C-7, C-11
receive/transmit
FIFO B-4
receiver
FIFO 2-3
overflow 5-14, B-3, B-5, B-12, C-2, C-8
pair 2-9, 4-11, 5-4
redundant
Rx/Tx fault 5-14
TxRx toggle C-1
transceivers 4-2, B-6
resolution bus switch 4-6
ring continuity 3-1, 5-25
ring
integrity 2-9, 3-5, 5-4, 5-10, 5-25
topology 2-4, 3-1
rotary switches 3-3, 3-5

S

SCRAMNet monitor 3-4
selected interrupt method 5-12
selected-interrupt method 2-6

self interrupt 2-6, 5-12
serial network E-1
setup 1-1, 2-5, 2-8, 2-10, 5-3, 5-12
shadow memory..... B-9, C-6
shared memory
  access..... B-13
  address..... B-13, C-9
shortwords..... 2-10, 5-15, 5-19, B-8
signal 1-1, 2-11, 3-4, 4-10, 5-22
signal error B-5
SIMMs..... 4-2, 4-6, 4-7
stretched cycle E-2, E-9, E-11
subroutine..... 2-1
switch..... 1-1, 2-10, 2-11, 3-1, 3-2, 3-4, 3-5, 4-1, 4-2, 4-5, 4-6, 4-11, 4-12, 4-13, 5-20, 5-25
  resolution bus..... 4-6
  selectable 2-1, 3-1
  status..... 3-5

T

temperature range A-1
throughput..... 2-9, 2-10, 5-6, 5-7, 5-19
  network..... 2-4, 2-10, 5-17
time-out..... 4-13, 4-14, 5-6, 5-7, B-5
timer
  global mode 2-8
  high-resolution mode..... 2-8
  sub-frame..... 2-8
transaction
  host..... E-2
  network..... E-11, E-12
transmit
  enable C-1
  FIFO5-26, 5-28, B-2, B-4, B-5, B-9, B-12, C-2, C-6
  interrupt event..... 5-6
  interrupts..... 2-5, 2-6, 2-8, 5-9, 5-10, 5-12
  retry time-out 5-14
  retry..... 5-14, B-5, C-2
transparent shared memory..... 5-3
trigger events B-6
trigger output signals..... 5-15
triggering..... 2-4

U

unaligned transfers..... 3-3
unpack..... 1-1
up-stream node 4-11

V

variable length message..... 2-4, 3-1, 3-2, 4-1, 4-7, 5-6, B-8, C-3
variable message length..... A-1, B-8, C-3
virtual page number C-10
virtual paging. 2-3, 3-2, 4-13, 5-1, 5-2, B-1, B-14, C-10
virtual-paging mode 5-2
VME
  cycle..... E-11
  holdoff mode 2-10
  holdoff..... 2-3, 2-10, 5-26, B-11, C-7, F-2
  interrupt priority level B-1, B-16, C-11
  P2 cabling option D-2
VME6U host interface..... 1-1, 3-1, 3-3
VMEbus 3-1, 3-2, 3-3, 5-26, A-1, A-4, B-10, C-6

W

weight..... A-1
wire loopback2-10, 2-11, 5-20, 5-21, 5-24, 5-25, B-4, B-7
  enable..... C-3
  mode 5-22, 5-23
worst case scenario E-6, E-7, E-8, E-9, E-11
write own slot enable 2-6
write-me-last mode..... 2-11, 5-28

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP