

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.

Find the *Pentek 4283* at our website: ***Click HERE***

OPERATING MANUAL

MODELS 4200A and 4201A

MIX Baseboard for VMEbus Systems

Master/Slave Interface

VSB Master Interface (Model 4200A)

Pentek, Inc.
One Park Way
Upper Saddle River, NJ 07458
(201) 818-5900

Copyright © 1993 - 2003

Manual Part No: 800.42000 Rev: A.2 - March 12, 2003

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Pentek Model 4200/01(A) Operating Manual
Revision History

Date	Rev	Applicable Serial #'s	Comments
06/15/93	Preliminary	9308006 - Forward	Initial product release
04/22/94	A	9416001 - Forward	Complete re-write. Expanded and revised.
06/10/94	A.1	9416001 - Forward	Added Section 3.2.6 (VIC/VAC register access from 68030), and added these address ranges to 68030 Address Map (Table 3-6). Added info on re-mapping Interrupt Vectors (Table 3-11) to Section 3.7.1.1. Changed number of table already in this section from 3-9 to 3-10. Clarify the difference between the three Host Monitor commands to run downloaded code in Section 2.4.14. Corrected MIX Flag Register address in Figs. 2-9 and 2-10, and in Tables 3-6, 3-8 and 3-9. Corrected grammatical error in Section 3.7.1.2. Added new Appendix (B) about Distribution Disk. Renamed previous Appendices B and C as C and D.
03/12/03	A.2	9416001 - Forward	Minimal update for 4200/01A.

WARRANTY

Pentek warrants that all products manufactured by Pentek conform to published Pentek specifications and are free from defects in materials and workmanship for a period of one year from the date of delivery when used under normal operating conditions and within the service conditions for which they were furnished.

The obligation of Pentek arising from a warranty claim shall be limited to repairing or at its option, replacing without charge, any product which in Pentek's sole opinion proves to be defective within the scope of the warranty.

Pentek must be notified in writing of the defect or nonconformity within the warranty period and the affected product returned to Pentek within thirty days after discovery of such defect or nonconformity.

Buyer shall prepay shipping charges, taxes, duties and insurance for products returned to Pentek for warranty service. Pentek shall pay for the return of products to buyer except for products returned from another country.

Pentek shall have no responsibility for any defect or damage caused by improper installation, unauthorized modification, misuse, neglect, inadequate maintenance, accident or for any product which has been repaired or altered by anyone other than Pentek or its authorized representatives.

The warranty described above is buyer's sole and exclusive remedy and no other warranty, whether written or oral, is expressed or implied. Pentek specifically disclaims fitness for a particular purpose. Under no circumstances shall Pentek be liable for any direct, indirect, special, incidental or consequential damages, expenses, losses or delays (including loss of profits) based on contract, tort, or any other legal theory.

Printed in the USA. All rights reserved. Contents of this publication may not be reproduced in any form without written permission.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table of Contents

Page

Chapter 1 - General Information

1.1 Introduction 7
1.2 Features 7
1.2.1 VMEbus and VSB Interfaces 7
1.2.1.1 System Controller 7
1.2.1.2 VMEbus Master Cycles 8
1.2.1.3 VMEbus Slave Cycles 8
1.2.1.4 Block Transfers 9
1.2.1.5 Interrupt Generation 9
1.2.1.6 Interrupt Handling 9
1.2.1.7 Local Bus Address Decoder 9
1.2.2 68030 Processor 9
1.2.3 MIX Bus Interface 10
1.2.4 Memory Resources 11
1.2.4.1 Static RAM 11
1.2.4.2 Flash EEPROM 11
1.2.5 Dual Serial I/O 11
1.3 Block Diagram 12
1.4 Specifications 12

Chapter 2: Installation, Connections and Setup

2.1 Inspection 17
2.2 Jumper Settings 17
2.2.1 VMEbus Slot 1 System Controller Jumper 17
2.2.2 VMEbus Slave Base Address Jumpers 20
2.2.4 Flash EEPROM Write Protect Jumpers 21
2.2.3 VSB System Arbiter Jumpers 21
2.2.5 Serial Port Connectivity Jumpers 22
2.3 The Model 4200/01 Front Panel 23
2.3.1 The Reset Switch 23
2.3.2 The Interrupt Switch 24
2.3.2.1 Restoring Factory Defaults with the Interrupt and Reset Switches 24
2.3.3 The LED Indicator 24
2.3.4 The External Port Header 24
2.3.4.1 Port Receive & Transmit Signals 25
2.3.4.2 External Interrupt Signal 25
2.3.4.3 Reset Signals 25
2.4 The 4200/01 Monitor Program 26
2.4.1 Connecting to the Serial Port 26
2.4.2 Default Communication Parameters 26
2.4.3 Booting the Monitor Program 27
2.4.4 Changing the VME Slave Base Address Table 29
2.4.5 Changing the 68030's Memory Map 32
2.4.6 VME Attributes 35
2.4.7 Examining and Modifying 4200/01 Memory Contents 37

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table of Contents (continued)

Page

Chapter 2: Installation, Connections and Setup (continued)

2.4 The 4200/01 Monitor Program (continued)
- 2.4.8 Loading Programs to the 4200/01 with the Monitor 38
  - 2.4.8.1 Loading Programs to the 4200/01's SRAM 38
  - 2.4.8.2 Loading Programs to the User/Configuration Flash EEPROM ... 39
  - 2.4.8.3 Copying EEPROM Code into SRAM 40
- 2.4.9 Starting and Stopping Programs from the Monitor 41
- 2.4.10 Upgrading 4200/01 Firmware 41
- 2.4.11 Selecting Interrupt Settings with the Monitor 44
- 2.4.12 Automatically Executing User Code 44
- 2.4.13 Issuing Other Commands with the Monitor 44
- 2.4.14 VMEbus Access to Monitor Features 44
  - 2.4.14.1 Host Interface Program - 4200.C 46
  - 2.4.14.2 Host Interface Program Interrupt Access 48
  - 2.4.14.3 Modifying A32 or A24 Base Addresses from VME A16 Space 49
  - 2.4.14.4 Changing the VSB Access Region Size from the VMEbus ... 50

2.5 Removing and Installing the Model 4200/01 Front Panel 51
- 2.5.1 Tools Required 51
- 2.5.2 Removing the 4200/01 Mezzanine Board 51
- 2.5.3 Removing or Installing the Front Panel Board Header 52
- 2.5.4 Removing the Front Panel 53
- 2.5.5 Installing the Front Panel 53
- 2.5.6 Replacing the Mezzanine Board 54

2.6 Installing MIX Expansion Modules on the 4200/01 54

Chapter 3: Operation

3.1 Operating Theory 55
- 3.1.1 The VME Interface Controller (VIC068A) 55
- 3.1.2 The VME Address Controller (VAC068A) 58
- 3.1.3 The 68030 58
- 3.1.4 SRAM Allocation 59
- 3.1.5 The MIX Bus Interface 59
- 3.1.6 4200/01 Data Cycles 60
- 3.1.7 Local Bus Arbitration 61

3.2 Model 4200/01 Memory Maps 61
- 3.2.1 VME Slave Memory Map 61
- 3.2.2 68030 Memory Map 64
- 3.2.3 The MIX Control Register 65
- 3.2.4 The MIX Flag Register 67
- 3.2.5 The MIX Status Register 69
- 3.2.6 Access to the VIC and VAC Registers from the 68030 70

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table of Contents *(continued)*

Page

Chapter 3: Operation *(continued)*

3.3 Modifying the Contents of the Flash EEPROMs 71
3.3.1 Saving Changes to the Address Tables 71
3.3.2 Writing Application Code into the Flash EEPROMs 74

3.4 Slave Operation 74
3.4.1 The 4200/01 as a VMEbus Slave 75
3.4.2 The 4200/01 as a MIX Bus Slave 75
3.5 Master Operation 76
3.5.1 The 4200/01 as a VMEbus Master 76
3.5.2 The 4200/01 as a MIX Bus Master 77
3.5.2.1 The 4200/01's MIX Interface as a VMEbus Master 77
3.5.2.2 The 4200's MIX Interface as a VSBus Master 77
3.5.3 The 4200 as a VSBus Master 78
3.5.4 DMA and the Model 4200/01 78
3.6 System Controller Operation 79
3.6.1 The 4200/01 as a VMEbus System Controller 79
3.6.2 The 4200 as a VSBus System Arbiter 79
3.7 Generating Interrupts with the 4200/01 79
3.7.1 The 4200/01 as a VMEbus Interrupter 79
3.7.1.1 Interrupting the VMEbus from a MIX Module 80
3.7.1.2 Interrupting the VMEbus from a 68030 Program 81
3.7.2 The 4200/01 as a MIX Bus Interrupter 81
3.7.2.1 Interrupting the MIX Bus from the VMEbus 82
3.7.2.2 Interrupting MIX Processor Modules from 68030 Programs .. 82
3.7.2.3 Local Conditions That Will Generate MIX Interrupts 82
3.8 Handling Interrupts to the 4200/01 84
3.8.1 The 4200/01 as a VMEbus Interrupt Handler 84
3.8.1.1 VMEbus Interrupts to the 68030 84
3.8.1.2 VMEbus Interrupts to the MIX Bus 84
3.8.2 The 4200/01 as a MIX Bus Interrupt Handler 84
3.8.2.1 MIX Bus Interrupts to the 68030 84
3.8.2.2 MIX Bus Interrupts to the VMEbus 85
3.8.3 The 4200 as a VSBus Interrupt Handler 85
3.8.3.1 VSBus Interrupts to the 68030 85
3.8.3.2 VSBus Interrupts to the MIX Bus 85
3.9 Additional Features of the Model 4200/01 85
3.9.1 Using the VAC's Programmable Timer 85
3.9.2 Using the VAC's UART 86
3.9.3 Split Bus Operation 87

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table of Contents (continued)

Page

Appendix A: Cypress Semiconductor VIC068A/VAC068A Register Maps and Definitions

A.1 VIC068A Register Map and Definitions A-3
A.2 VAC068A Register Map and Definitions A-35

Appendix B: The 4200/01 Distribution Diskette

B.1 Introduction B-1
B.2 Contents of 4200.zip B-1
B.3 Utility Programs for the Model 4200/01 ..... B-5

Appendix C: MIX Tutorial

C.0 Introduction C-1
C.1 MIX Baseboards C-1
C.2 MIX Modules (Expansion Modules) C-2
C.3 MIX Baseboard Connector C-4
C.4 MIX Stiffener C-4
C.5 Flathead Screws C-4
C.6 First Slot (Nested) Module Installation C-5
C.7 Installing Ejector Handles on the MIX Module Front Panel C-6
C.8 MIX Stacking Connector C-7
C.9 MIX Jackscrews C-7
C.10 MIX Spacer Board C-7
C.11 Second Slot Module Installation C-8
C.12 Adding a Second Expansion Module C-8
C.13 Adding a Third Expansion Module C-9
C.14 Installing the Assembly into the VMEbus Card Cage C-10
C.15 A Glossary of MIX Terms C-11

Appendix D: Operating Note - A24 Slave Addressing

D.1 Introduction D-1
D.2 A24 Address Decoding and the VAC068A D-1
D.3 Case 1: No P2 Connectors D-1
D.4 Case 2: The Bus Master Does Not Drive the P2 Address Lines D-2
D.5 Case 3: The Upper Address Lines are Driven by the Bus Master D-2
D.6 Flash EEPROM Address Conflict D-2

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Chapter 1 - General Information

1.1 Introduction

The Pentek Models 4200A and 4201A convert Pentek's full complement of MIX modules into standard 6U single-slot VMEbus boards with full bus master capability. The Model 4200A features three bus interfaces; VME, VSB (VME Subsystem Bus) and MIX. Master and slave operations are supported by both the VME and MIX interfaces, while the VSB interface offers master operation only. The Model 4201A is the same as the Model 4200A, but without the VSB interface.

The MIX Bus is a local daughter card bus that accepts one, two or three MIX daughter card modules in a stacking arrangement. Each module requires one slot in the VME card cage.

1.2 Features

The following subsections summarize and briefly describe the features of the Pentek Model 4200A and Model 4201A VME/MIX Baseboards.

1.2.1 VMEbus and VSB Interfaces

The VMEbus interface is implemented with the Cypress Semiconductor VIC068A VMEbus Interface Controller (VIC) and the VAC068A VMEbus Address Controller (VAC) chip set, providing well-defined and fully-characterized VMEbus signal timing. These devices fully meet the requirements of the IEEE VMEbus Specification 1014 Rev C.1.

The VIC/VAC set is designed to be highly compatible with the 68030, which serves as the initialization and control resource for these devices. The combination of these three elements (68030/VIC/VAC) represents an extremely popular and standard configuration, found on hundreds of VMEbus products, and available from dozens of manufacturers.

The VME Subsystem Bus (VSB) is a secondary bus utilizing the outer rows of pins on the VME P2 connector, which are unused in the standard VMEbus implementation. The Model 4200A operates as a VSB master and interrupt handler, via the VSB1400A/B chip set from PLX. There is no VSB interface on the Model 4201A.

1.2.1.1 System Controller

Because of its general purpose architecture, the Model 4200/01A can serve as a complete VMEbus system controller in the VMEbus chassis.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1.2 Features (continued)

1.2.1 VMEbus and VSB Interfaces (continued)

1.2.1.1 System Controller (continued)

Within the 4200/01A, the VIC supports priority and round robin arbitration schemes; drives the IACK daisy-chain and the Bus Grant I/O daisy-chain; drives the SYSCLK and SYSRESET lines; and provides VMEbus arbitration and transfer timeout timers.

1.2.1.2 VMEbus Master Cycles

All phases of VMEbus mastership are supported when access to the VMEbus is requested by the 68030, the VIC/VAC DMA controller, or MIX Bus master devices. Additional bus master features are: data transfers in A32, A24 or A16 address space, utilizing D32, D16, or D08 data widths; full address modifier (AM) code generation; bus requests on all four levels; Release on Request, Release When Done, Release on Clear, Release under RMC, and Bus Capture and Hold requester capabilities; Indivisible Read-Modify-Write and Multiple-Address cycles; Deadlock and self-access handling provisions; and Master Write-Posting (latches data from the local bus and releases it to be free from VMEbus arbitration delays).

1.2.1.3 VMEbus Slave Cycles

Slave access from the VMEbus is supported in many different configurations, providing memory mapping of each MIX module, the local bus memory resources and the VSB. Slave support features include: data transfers in A32, A24 or A16 address space, utilizing D32, D16, or D08 data widths; full address modifier (AM) code handling; Block Transfers, programmable for both accelerated and non-accelerated modes; programmable Physical Address Strobe and data acquisition delays; and Slave Write-Posting (latches data from the VMEbus and releases it to be free from local bus arbitration delays).

Two separate slave base address registers within the VAC allow the Model 4200A to respond to two different slave addresses. The size of data words is separately programmable for the A16 and A24 address spaces, i. e., it is possible for one slave address to be programmed for A24/D16 access, and the other for A16/D32.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1.2 **Features** (continued)

1.2.1 **VMEbus and VSB Interfaces** (continued)

1.2.1.4 Block Transfers

Both master and slave block transfers are supported with extended addressing (32-bit incrementing address) provided by the VAC. Every time a 256-byte boundary is crossed, a new Address Strobe is asserted to maintain VMEbus compliance. Other Block Transfer-related features include Master block transfers with local DMA, MOVEM-type block transfers, and Slave transfers in accelerated or non-accelerated mode.

1.2.1.5 Interrupt Generation

All seven interrupts levels of the VMEbus can be generated and a user defined interrupt vector is supported for each level. Interrupts can also be generated to the 68030 for various VMEbus transfer error conditions including ACFAIL, SYSFAIL, BERR and bus arbitration timeout.

1.2.1.6 Interrupt Handling

The VIC handles interrupts from the following sources: all seven levels of the VMEbus; all three MIX modules on the MIX bus; the VSB interrupt; four interrupts from the two VAC serial ports; status and control interrupts from the VAC; and seven interrupts from the 68030. Also supported are edge/level control, polarity control, and user defined local IPL (Interrupt Priority Level).

1.2.1.7 Local Bus Address Decoder

The VAC decodes nearly all of the address space within the Model 4200/01A. This provides a very flexible allocation of memory within the onboard SRAM, Flash EEPROM memories, the VSB interface, and the MIX interface.

1.2.2 **68030 Processor**

The Model 4200A and Model 4201A feature an onboard 40 MHz 68EC030 processor that acts as the interface controller and DMA engine. A 1 MB Static RAM (optionally 4 MB) and two 256 kB EPROMs service the 68030 via the local bus. The 68030 has full memory access to the VMEbus, the VSB and the MIX bus, with 32-bit data and 32-bit address support for all transfers.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1.2 Features (continued)

1.2.2 68030 Processor (continued)

Since the DMA controller within the VIC/VAC chip set can only move data between the VMEbus and the local bus, data moves between two local bus resources (e. g., the SRAM and the MIX bus) are handled by the 68030 using DMA-like routines stored in the User/Configuration EEPROM memory. In this way, the 68030 acts as a flexible, multi-channel DMA controller. Source address, destination address, transfer type and block length parameters are accepted for each DMA channel. Transfers can be initiated by any of the numerous interrupts processed by the 68030. These include interrupts from the MIX bus, the VMEbus, the VSB, the serial ports, etc.

1.2.3 MIX Bus Interface

This interface provides a bidirectional 32-bit data path between the local bus and the MIX Bus. Full memory-mapped access to all modules on the MIX bus is supported from both from the 68030, for DMA transfers, and from the VMEbus. This allows the MIX modules to appear as VMEbus slave devices.

MIX modules with MIX Bus master capability use the Bus Interface to access the VMEbus and the VSB with A32 D32 bus master capabilities. For example, when combined with the Model 4200A, the Model 4257 Dual TMS320C40 Co-Processor MIX module becomes a complete Dual TMS320C40 Processor VME Master/Slave board, all in a single 6U VME card cage slot.

The MIX bus can also be used for dedicated high-speed data transfers between MIX modules. These transfers can be initiated by any MIX module having MIX bus master capability or by the 68030 acting as a DMA controller. It is even possible to move data between MIX modules by issuing read and write commands from the VMEbus.

An important feature of the Model 4200/01A is its ability to decouple the MIX bus from the local bus, allowing each to conduct memory cycles independent of the other. Only when access across this interface is required must one bus arbitrate for the other. With the write-posting support of the VIC/VAC chip set, transfers between MIX and VME busses are enhanced even further.

When the MIX and local busses are decoupled, a set of bidirectional registers is provided in the MIX/local Interface that act like mailbox registers between the busses. A write from one bus to the transport register occurs without bus arbitration, and causes a maskable interrupt to the other bus. The other bus may read this register without arbitrating for the generating bus, and perform some appropriate action.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1.2 **Features** (continued)

1.2.3 **MIX Bus Interface** (continued)

As a MIX Bus Master, the Models 4200A and 4201A are also capable of handling Interrupts from any expansion modules on the MIX Bus stack.

1.2.4 **Memory Resources**

Two different types of memory devices are used in the Model 4200/01A, Static RAM (SRAM), and Flash Electrically Erasable Programmable ROM (EEPROM).

1.2.4.1 Static RAM

The local bus serves as the central data path for all transfers within the unit. A fast 1 MB Static RAM provides zero-wait-state operation as a stack, and as a general purpose program and data staging area, for the 68030. The SRAM may optionally be expanded to 2 MB or 4 MB.

1.2.4.2 Flash EEPROM

There are two 256 kB Flash EEPROM memories contained on the 4200/01A. These non-volatile devices can be written to without removing them from the board. One is used for the dedicated factory supplied firmware to initialize the board and to support the VIC/VAC functions.

The second Flash EEPROM is available to the user to support powerful, custom embedded controller applications. This can include complex or compound DMA routines, as well as complete standalone applications.

1.2.5 **Dual Serial I/O**

The VAC contains a dual, full duplex UART, with programmable baud rate selections between 300 and 9600 and support both 7 and 8 bit characters with programmable parity checking and generation. The lines are translated to RS-232C levels and brought to a pair of front panel DB-25 type connectors. If the front panel is removed, the serial ports are accessed via a 16-pin dual inline header on the circuit board.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1.3 Block Diagram

A simplified block diagram of the Pentek Model 4200/01A VME/MIX Baseboard is presented on the following page. More thorough diagrams and detailed discussion of the operating characteristics of the board can be found in Chapter 3 of this manual.

1.4 Specifications

Processor:

Type:	68EC030
Clock Speed:	40 MHz
Address Bus Width:	32 Bits
Data Bus Width:	32 Bits

Memory:

SRAM:

Size:	1 MB (256k x 32), Optionally expandable to 2 MB (512k x 32) or 4 MB (1M x 32)
Access:	68030, VMEbus Masters, MIX Bus Masters

Flash EEPROM:

User/Configuration PROM:

Size:	256 kB (64k x 32)
Access:	68030

Monitor PROM:

Size:	256 kB (64k x 32)
Access:	68030, VMEbus Masters, Mix Bus Masters, Serial Ports

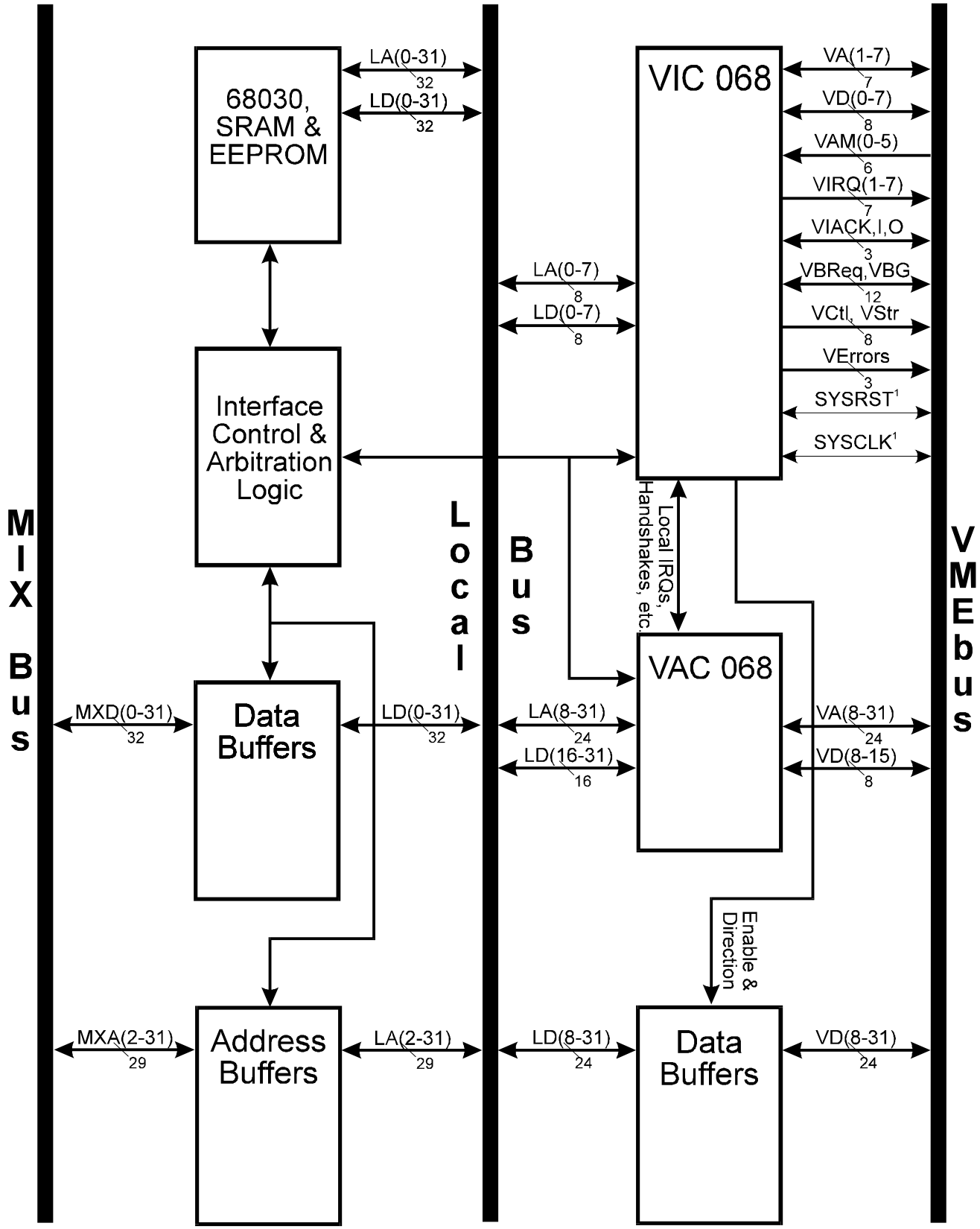
VMEbus Interface:

Master Functions:

Addressing Capabilities:	A32, A24 or A16	
Data Transfer:	D32:	Block Transfer (BLT) Read-Modify-Write(RMW) Unaligned Transfer (UAT)
	D16:	Block Transfer (BLT) Read-Modify-Write (RMW)
	D08:	Even or Odd Byte Positioning (EO) Read-Modify-Write (RMW)
Bus Request Levels:	All 4 VME Levels (BREQ0 - BREQ3)	

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Figure 1-1: Pentek Model 4200/4201A - Top Level Block Diagram



1 - These lines are outputs if 4200/01 is configured as slot 1 controller, inputs otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1.4 Specifications (continued)

VMEbus Interface (continued):

Master Functions (continued):

Bus Release Modes:		Release on Request (ROR) Release When Done (RWD) Release on Clear (ROC) Release under RMC* control Bus Capture and Hold (BCAP) Fair-Access Timeout (FAIR) Master Write-Posting Indivisible Multiple-Address Cycles (IMACs)
Address Modifiers:		Full Encoding Capability
Interrupt Handling:	D08:	Odd Byte Positioning Handles all 7 VME levels (IH 1-7) Supports Mailbox Interrupts

Slave Functions:

Address Spaces:		A32, A24 or A16
Data Transfer:	D32:	Block Transfer (BLT) Read-Modify-Write(RMW) Unaligned Transfer (UAT)
	D16:	Block Transfer (BLT) Read-Modify-Write (RMW)
	D08:	Even or Odd Byte Positioning (EO) Read-Modify-Write (RMW) Supports Slave Write-Posting
Block Transfer Support:		Programmable (DMA, Non-DMA, None)
Address Modifiers:		Full Decoding Capability
Interrupt Generation:	D08:	Odd Byte Positioning Interrupts on all 7 VME levels (I 1-7) Release on Acknowledge (ROAK) Supports Mailbox Interrupts

System Controller Functions:

Arbitration:	Round-Robin Sequence (RRS) Prioritized (PRI) Programmable Timer for Arbitration Timeout
Bus Lines Driven:	SYSCLK SYSRESET Bus Grant I/O Daisy Chain (all 4 levels) IACK Daisy Chain

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

1.4 Specifications (continued)

VME Subsystem Bus (VSB) Interface:

Master, Interrupt Handler

Address Bus Width:	32 Bits
Data Bus Width:	32 Bits

MIX Bus Interface: Master/Slave, Interrupt Handler

Address Bus Width:	32 Bits
Data Bus Width:	32 Bits

DMA Transfer Rates:

Bus Master	Transfer Path	Speed
VIC/VAC	VME - SRAM	16 MB/Sec
68030	MIX - SRAM	3 MB/Sec
	VSB - MIX*	3 MB/Sec
	SRAM - VSB*	3 MB/Sec
	* - Single cycle access only	
MIX Module (Upper Master)	Consult the Operating Manual for the MIX Module in question (4254, 4257 or 4270)	

VME Slave Delay (DS* - DTACK*):

SRAM:	Single Cycle: 200 nsec Block Transfer: 100 nsec
MIX:	Single Cycle: 300 nsec Block Transfer: N/A

Serial Ports (Full Duplex RS-232):

Number of Ports:	2
BAUD Rates:	(Common to both ports) 300, 600, 1200, 2400, 4800 or 9600. Non-standard rates also supported.
Parity:	Odd, Even, None
Stop Bits:	2 on Transmit, 1 on Receive
Buffering:	Double on Transmit, Quint on Receive
Interrupt-Driven Operation:	Supported

Power: 1A typ. @ +5V

Size: Standard 6U VME Board

Depth:	160 mm (6.3 in.)
Height:	233.5 mm (9.2 in.)
Width:	With front panel installed, 0.8 in.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

This page is intentionally blank

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Chapter 2: Installation, Connections and Setup

2.1 Inspection

After unpacking the unit, inspect it carefully for possible damage to connectors or components. If you discover any damage, please contact Pentek immediately, at the phone number shown on this manual's title page. Please save the original shipping container and packing material, in case re-shipment is required.

2.2 Jumper Settings

On the two following pages are drawings of the three printed circuit boards that comprise the Model 4200/01A, showing the locations of the Jumper Blocks referred to in the sections below. We refer to the larger board, with the VME backplane connectors (shown in Figure 2-1), as the VME board. The smaller board, which houses the 68030, its SRAM, and the Model 4200A's VSB Interface chip set (shown in Figure 2-2), is known as the Mezzanine board. All the jumper blocks utilized on these boards are of the 3-pin, single inline, right-angle variety, and are located near the edges of the boards. This allows access to the Jumper Blocks without removing the Mezzanine board from the VME board.

Additionally, there are two 6-pin dual inline Jumper Blocks on the Model 4200/01A's front panel board, for configuring the connections to the Serial Ports. A drawing of this board is also shown in Figure 2-2, showing the locations of these Jumper Blocks. Their functions are also described below.

2.2.1 VMEbus Slot 1 System Controller Jumper - JB1, VME Board

The Model 4200/01A's VME System Controller functions are enabled by the installation of a shorting jumper between pins 1 and 2 of Jumper Block JB1 on the VME board. Installing a jumper in this position allows the 4200/01A to arbitrate requests from VMEbus Masters (the default arbitration mode is RRS, or Round Robin Sequenced), and to drive the SYSCLOCK and SYSRESET lines, and the IACK and Bus Grant I/O daisy chains, on the VMEbus.

To disable the Model 4200/01A's System Controller functions (i. e., if you wish to use another device as your Slot 1 System Controller), place a shorting jumper between pins 2 and 3 of JB1 on the 4200/01A's VME board. The factory default settings have the jumper in this position. Remember that there can be ONE AND **ONLY** ONE System Controller in any given VME card cage, and that the controller MUST reside in Slot 1 of the card cage.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

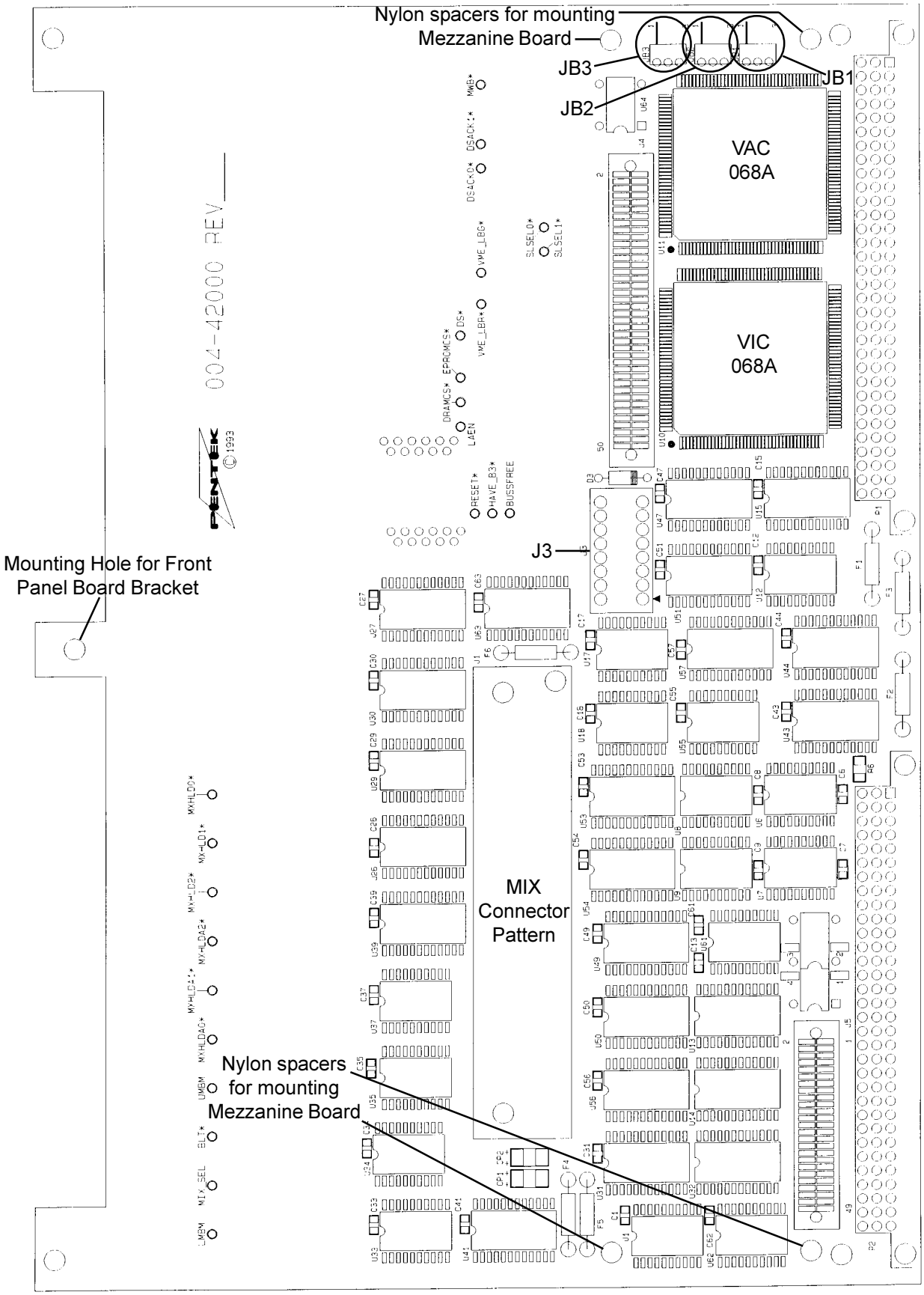


Figure 2-1: Model 4200/4201A VME board drawing showing jumper block and mounting hole locations

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

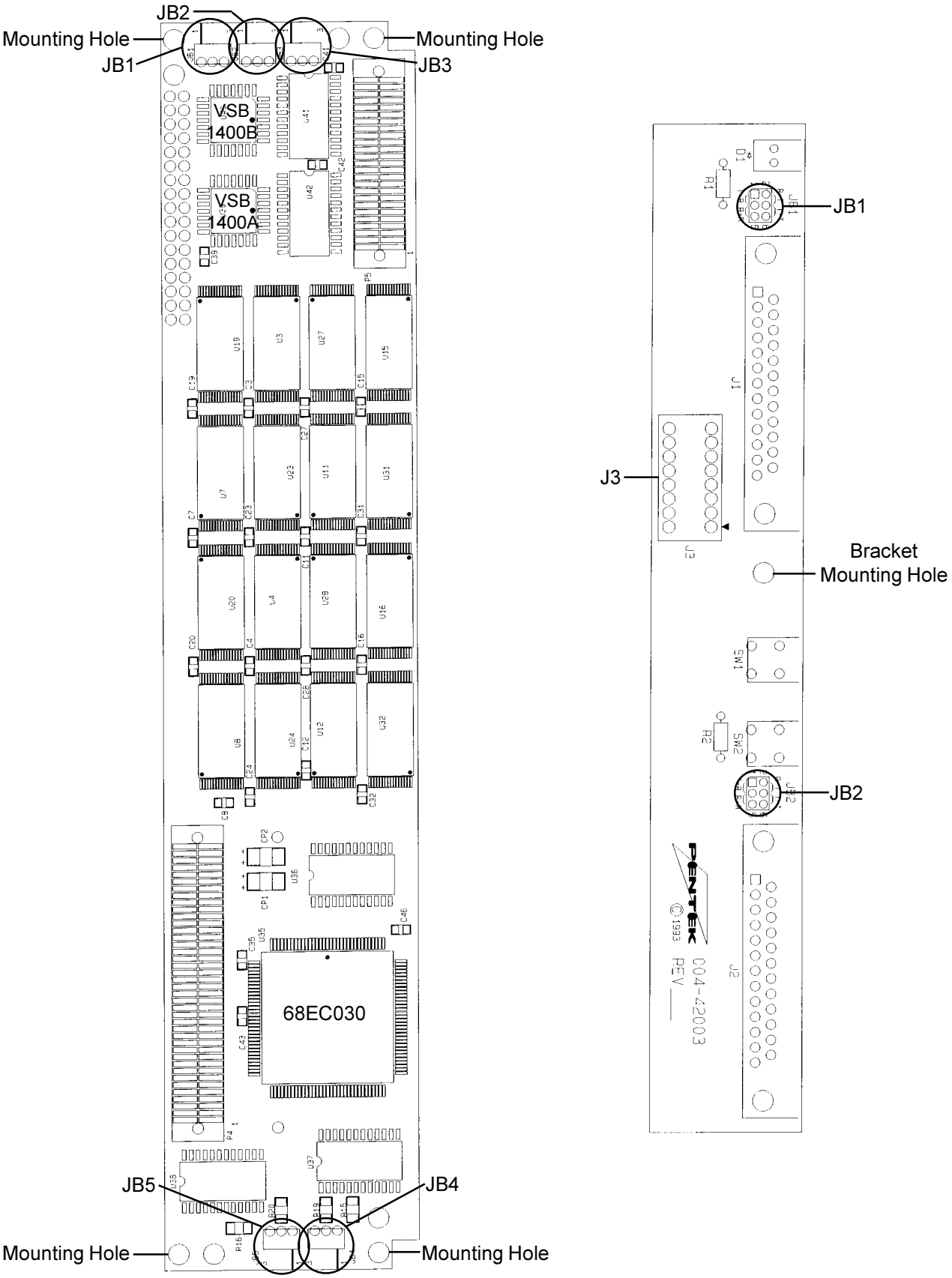


Figure 2-2: Model 4200/4201A Mezzanine board (left) and Front Panel board (right) drawings showing jumper block and mounting hole locations. The side of Mezzanine board shown is mounted face down.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.2 Jumper Settings (continued)

2.2.1 VMEbus Slot 1 System Controller Jumper (continued)

Table 2-1, below summarizes the System Controller Jumper Block settings.

Table 2-1
Pentek Model 4200/01A
VME System Controller Jumper Block
JB1, VME Board

Jumper Position	VME System Control
1 - 2	Enabled
*2 - 3	Disabled
* - Factory Default Setting	

2.2.2 VMEbus Slave Base Address Jumpers - JB2 and JB3, VME Board

At power-up and during resets, the 68030 reads the state of two Jumper Blocks on the VME board, JB2 and JB3. The result of that operation is used to select one of four default base address configurations for the 3 VME Slave resources, from a table in the User/Configuration Flash EEPROM. The Slave resources are the Inter-Processor Communication Flag (ICF) Register, which resides in A16 space, and the SRAM and MIX bus, which may reside in A24 or A32 space. The factory default settings place a jumper between pins 1 & 2 of JB3 (selecting A24 space for the SRAM and MIX bus), and between pins 2 & 3 of JB2. Table 2-2, below describes the default configurations.

Table 2-2
Pentek Model 4200/01A
Default VME Slave Base Address Table
JB2 and JB3, VME Board

Jumper Positions		SRAM & MIX Address Space	SRAM Base Address	MIX Base Address	ICF Regs (A16)
JB3	JB2				
2 - 3	2 - 3	A32	0x0000 0000	0x4000 0000	0x0000
2 - 3	1 - 2	A32	0x8000 0000	0xC000 0000	0xA000
*1 - 2	2 - 3	A24	0x00 0000	0x40 0000	0x8000
1 - 2	1 - 2	A24	0x80 0000	0xC0 0000	0xC000
* - Factory Default Settings					

Functions are provided in the firmware Monitor program to change the contents of the Base Address table in the User/Configuration Flash EEPROM. See Sections 2.2.4 and 2.4.4, later in this chapter, for more details.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.2 Jumper Settings (continued)

2.2.3 VSB System Arbiter Jumpers - JB1, JB2 and JB3, Mezzanine Board (4200 only)

The VSB Interface on the Model 4200A can be configured to include the system bus arbitration function. This is accomplished by placing shorting jumpers between pins 2 and 3 of JB1, JB2 and JB3 on the 4200A's Mezzanine board. If you have configured the 4200A as the VME Slot 1 Controller (see Section 2.2.1), then you should also configure it as VSB Arbiter.

If you would rather have another device in your card cage handle VSBus arbitration, then the 4200A's VSB arbiter functions should be disabled. Do this by placing the shorting jumpers between pins 1 & 2 of JB1, JB2 and JB3 on the 4200A's Mezzanine board. This is the factory default setting.

Note that the shorting jumpers should always be in the same positions on all three of these Jumper Blocks. Table 2-3, below, summarizes the settings.

None of these three Jumper Blocks is included on the Model 4201.

Table 2-3
Pentek Model 4200A
VSB System Controller Jumper Blocks
JB1, JB2, and JB3, Mezzanine Board

Jumper Positions	VSB System Arbitration
*1 - 2	Disabled
2 - 3	Enabled
* - Factory Default Setting	

2.2.4 Flash EEPROM Write Protect Jumpers - JB4 and JB5, Mezzanine Board

As mentioned above, in Section 2.2.2, the Flash EEPROMs (both the User/Configuration PROM and the Monitor PROM) can be re-programmed in-circuit. This feature is enabled by installing a shorting jumper between pins 1 and 2 of Jumper Blocks JB4 (for the User/Configuration EEPROM) and JB5 (for the Monitor EEPROM). This is the factory default setting, which Write-Enables the Flash memory. Placing the jumper between pins 2 and 3 of either of these Jumper Blocks will disable the re-programming feature, or Write-Protect the EEPROM in question.

Table 2-4, at the top of the next page, summarizes the settings of the Flash EEPROM Write Protect Jumper Blocks.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.2 Jumper Settings (continued)

2.2.4 Flash EEPROM Write Protect Jumpers (continued)

Table 2-4
Pentek Model 4200/01A
Flash EEPROM Write Protect Jumpers
JB4 and JB5, Mezzanine Board

Jumper Block	Flash EEPROM	Jumper Position	Write Function
JB4	User/ Configuration	*1 - 2	Enabled
		2 - 3	Disabled
JB5	Monitor	1 - 2	Enabled
		*2 - 3	Disabled
* - Factory Default Settings			

2.2.5 Serial Port Connectivity Jumpers - JB1 and JB2, Front Panel Board

Users familiar with RS-232 Serial Ports have no doubt been faced in the past with the quandary over whether the cable used with a given port should be configured in the "null modem" or "straight-through" modes. The Serial Port connectors on the Model 4200/01A's front panel board can be configured by means of jumper blocks to work with either kind of cable.

Jumper Block JB1 is used to determine the connectivity of Serial Port A, and JB2 configures Serial Port B. For both Jumper Blocks, a shorting jumper from pin 1 to pin 3 will make pin 3 of that port's DB-25 connector the Transmit line. A jumper from pin 2 to pin 4 on the block will make pin 2 of the DB-25 connector the Receive line. These are the factory default settings for these jumper blocks. To make pin 2 of the DB-25 connector the Transmit line, place a shorting jumper between pins 4 and 6 of the Jumper Block. A jumper from pin 3 to pin 5 on either of these blocks will make pin 3 on the corresponding port's DB-25 connector the Receiver. These settings are summarized in Table 2-5, on the following page.

With regard to the DB-25 serial port connectors, aside from the jumper programmable pins described above, pins 1 and 7 are Grounds, and all other pins are unused. For more information about the 4200/01A's front panel and Serial Ports, please refer to Sections 2.4.1 and 2.4.2, later in this chapter.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.2 Jumper Settings (continued)

2.2.5 Serial Port Connectivity Jumpers (continued)

Table 2-5
Pentek Model 4200/0A1
Serial Port Connectivity Jumpers
JB1 and JB2, Front Panel Board

Jumper Block	Serial Port	Jumper Position	DB-25 Connection
JB1	A	*1 - 3	Pin 3 - Transmit
		3 - 5	Pin 3 - Receive
		*2 - 4	Pin 2 - Receive
		4 - 6	Pin 2 - Transmit
JB2	B	*1 - 3	Pin 3 - Transmit
		3 - 5	Pin 3 - Receive
		*2 - 4	Pin 2 - Receive
		4 - 6	Pin 2 - Transmit
* - Factory Default Settings			

2.3 The Model 4200/01A Front Panel

The Models 4200A and 4201A are shipped with a front panel installed, providing easy access to the two RS-232 Serial Ports (via standard DB-25 connectors), reset and interrupt push-button switches, and an LED indicator. However, if space within your VME card cage is a scarce commodity, you may find it necessary to remove the baseboard's front panel, so that it may share a card cage slot with the first module in the MIX stack. Should you find this action to be necessary, we recommend that you keep the front panel installed on the Model 4200/01A until you have completed the initial configuration of the unit. These procedures are considerably easier to accomplish with the aid of the monitor program, run from the front panel serial port.

Instructions for "nested" installation of MIX modules may be found in Appendix C of this manual. The front panel switches and LED will be discussed in the subsections below. The remainder of this Chapter deals with using the firmware Monitor program, via Serial Port A on the front panel, in order to configure the 4200/01A to your particular needs. A section is also included covering the removal and installation of the panel.

2.3.1 The Reset Switch

Pressing the Reset push-button on the 4200/01A's front panel will issue a Global Reset. This is identical to the reset issued at power-up. If the 4200/01A is configured as the VME Slot 1 System Controller, a VME SYSRESET will also be issued when this button is pushed.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.3 The Model 4200/01A Front Panel (continued)

2.3.2 The Interrupt Switch

When the Interrupt push-button on the 4200/01A's front panel is pressed, the PIO9 pin on the VAC is pulled low. Depending upon how the VAC is configured, this input is mapped to either PIO7, PIO10, or PIO11, all of which are connected to Local IRQ inputs on the VIC. These inputs, depending upon the VIC's configuration, may either cause a VME IRQ to be issued, or activate one or more of the 68030's three IPL (Interrupt Pending Level) signals. Therefore, this switch may be used to issue a non-maskable interrupt to either the VMEbus or the 68030.

2.3.2.1 Restoring Factory Defaults with the Interrupt and Reset Switches

The Interrupt button and Reset button can be used in tandem to undo some changes made to the 4200/01A's Flash EEPROMs. This is accomplished by pressing the Interrupt button, and holding it in while you press and release the Reset button. Continue holding the Interrupt switch until the Monitor has finished printing its Logo to the terminal. See Section 2.4 for information about using the Monitor program to make changes to the Flash memory, and Section 2.4.4 for an example of the Logo.

2.3.3 The LED Indicator

The cathode of the LED on the 4200/01A's front panel is also connected to the PIO9 pin on the VAC. This pin can function as either an input (as it does when the Interrupt button is pressed) or a tri-state output. The LED's anode is pulled up to +5V via a resistor. Thus, the LED can be turned on by driving the VAC's PIO9 pin low (note that the LED lights when the Interrupt switch is pressed). A code segment to flash the LED is provided as one of the software examples on the enclosed diskette, described in Appendix B of this manual.

2.3.4 The External Port Header

The cable from the 4200/01A's front panel board is connected to J3 on the 4200/01A's VME board, which is located underneath the Mezzanine board. This connector can be used to access the serial port signals and all of the other functions described above when the front panel is removed. The physical implementation of this connector is a series of low-profile insertion pins arranged in a DIP-16 pattern. The pinout of this connector is given in Table 2.6, at the top of the next page. Signal functions are described in the subsections starting below the table.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.3 The Model 4200/01A Front Panel (continued)

2.3.4 The External Port Header (continued)

Table 2-6
Pentek Model 4200/01
External Port Connector Pinout

Signal	Pin #	Pin #	Signal
Ground	1	16	Port A Receive
Port B Receive	2	15	Port A Transmit
Port B Transmit	3	14	Ground
Ground	4	13	Ground
External Interrupt	5	12	Ground
Ground	6	11	Ground
Internal Reset	7	10	Soft Reset
Ground	8	9	Global Reset

2.3.4.1 Port Receive & Transmit Signals

These are the RS-232 level serial port communication lines. When the front panel board is installed, these signals are connected to the DB-25 Connectors via Jumper Blocks JB1 and JB2.

2.3.4.2 External Interrupt Signal

This is the line that the Interrupt switch on the front panel board is connected to. It is, in turn, connected to the PIO9 pin on the VAC068. See section 2.3.2, above, for further details.

2.3.4.3 Reset Signals

The three Reset lines available on the External Port Connector are used to reset different sub-circuits on the Model 4200/01A. The Global Reset is the line that is connected to the Front Panel Reset Button. When this line is pulled low, all circuitry on the 4200/01A is returned to it's initial, power-up state. If the Model 4200/01A is configured as the VME Slot 1 System Controller, then a VME SYSRESET will be asserted whenever a Global Reset is invoked. See Appendix A, regarding the VIC068A and VAC068A registers, for further details.

The Internal Reset line, when pulled low, will initiate a VIC Internal Reset sequence. This reset sequence does not change the states of any of the registers in the VAC, but completely initializes the VIC and the 68030. If the 4200/01A is your VME System Controller, a SYSRESET will also be asserted by the Internal Reset sequence. For a more detailed description of the Internal Reset, see the section in Appendix A referring to the VIC068A registers.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.3 The Model 4200/01A Front Panel (continued)

2.3.4 The External Port Header (continued)

2.3.4.3 Reset Signals (continued)

The Soft Reset line, when pulled low, will initiate a VAC Soft Reset sequence. This reset sequence does not change the states of any Internal VIC, VAC or 68030 registers, nor does it affect the memory map configuration. The function of the Soft Reset is to mask all pending interrupts associated specifically with the VAC. Specifically, this reset will mask the interrupts associated with the Serial Ports, the VAC's internal timer and the front panel Interrupt Button. For further information, see the section in Appendix A referring to the VAC068A registers.

2.4 The 4200/01A Monitor Program

The easiest and most direct path to tailoring the 4200/01A's features to the requirements of your application is via the use of the firmware Monitor program, run over the serial port. The Monitor provides features that allow modification of the VME Slave and 68030 address tables, and customization of details of the VME interface such as the Bus Request Level, the Bus Arbitration mode, etc. Features are also provided to allow the reading and modification of any memory location, or to download a program to the SRAM and run it. Additionally, the Monitor provides a means by which the 4200/01A's firmware may be modified or upgraded. Details on all the features described above are found in the sections that follow.

2.4.1 Connecting to the Serial Port

The Monitor initializes Serial Port A for its user interface. This port is available on the upper DB-25 connector on the Front Panel. The port may be connected to either an RS-232 "dumb" terminal, or to an RS-232 serial port on any PC or workstation capable of running terminal emulation software.

There are literally hundreds of communications software packages available which provide support for local RS-232 terminal emulation. Your selection should be dictated by the requirements of your hardware platform and your personal preferences.

2.4.2 Default Communication Parameters

The Monitor firmware initializes the 4200/01A's serial port communications parameters as listed in Table 2-7, at the top of the next page. Set your terminal or your emulation software accordingly.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.3 The Model 4200/01A Front Panel (continued)

2.4.2 Default Communication Parameters (continued)

Table 2-7
Pentek Model 4200/01
Default Serial Port
Communications Parameters

Baud Rate:	9600
Parity:	None
Data Bits:	8
Stop Bits:	1

Please remember that the two most common reasons for inability to communicate over any serial port are (a) incorrect settings of the communication parameters, and (b) crossed Receive and Transmit lines in the serial cable. If you are certain that your settings match those shown above and you are still unable to communicate with the 4200/01A over the serial port, try reversing the settings on Jumper Block JB1 on the Front Panel Board, to change the Transmit and Receive connections on Serial Port A. Refer to Section 2.2.5, above, for details.

2.4.3 Booting the Monitor Program

To boot the 4200/01A's Firmware Monitor Program, simply insert the board into the VME card cage, turn the power on, and press the Reset button on the front Panel.

CAUTION!!

Never insert the Model 4200/01 into the VME card cage with the power turned on!

After the Reset button is pressed, the Pentek Monitor logo and prompt should appear on your terminal screen, showing the amount of installed SRAM and the default VME slave addressing. A sample of the logo is shown in Figure 2-3, on the next page.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.3 Booting the Monitor Program (continued)

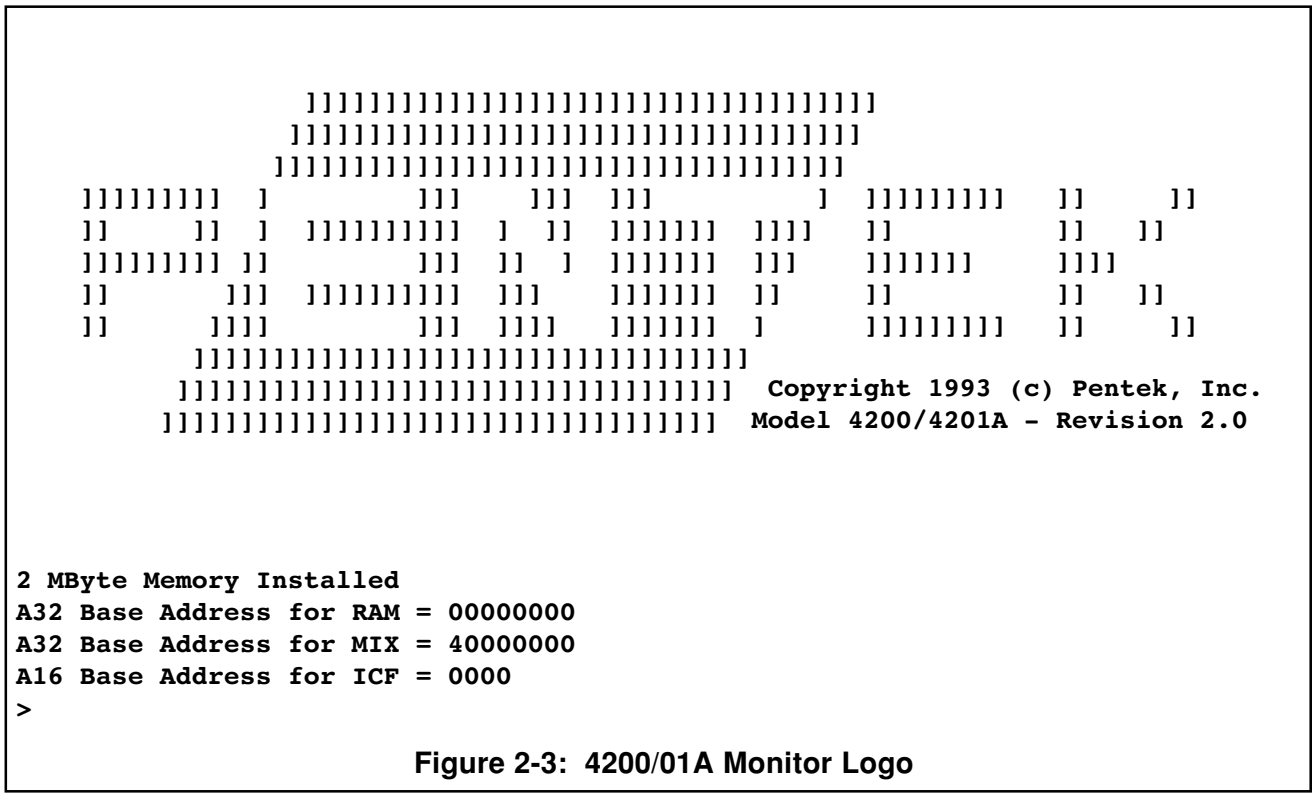


Figure 2-3: 4200/01A Monitor Logo

Press the **Return** or **Enter** key on your terminal at the **>** prompt for the Monitor menu, shown in Figure 2-4, below. The sections that begin on the following page describe the menu options.

```
Enter
 1 to Examine Memory
 2 to Upgrade Firmware
 3 to Download Program
 4 to Run Program
 5 to Stop Program
 6 to Load Program to User Flash EEProm
 7 to Copy Program from User Flash EEProm
 8 to Select VME Base Address Settings
 9 to Select Internal Memory Map
10 to Select VME Attributes
11 to Select Interrupt Settings
12 to Select Other Parameters
13 to Select Other Commands
>
```

Figure 2-4: 4200/01A Monitor Main Menu

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.3 Booting the Monitor Program (continued)

NOTE: Most of the Monitor's functions involve the making of changes to information stored in the User/Configuration Flash EEPROM. In order for these changes to be saved to the Flash Memory, the Write Protect Jumper for this device (JB4 on the Mezzanine board) must be placed in the Write Enable position (between pins 1 and 2), or the programming operations described in the sections below will fail. In the current revision of the firmware, the Monitor program will hang if a write operation is attempted to a protected EEPROM. If this should occur, you may restart the Monitor by pressing the Reset button on the front panel, or by cycling power to the card cage. See Section 2.2.4 for more information about the Write Protect Jumpers.

2.4.4 Changing the VME Slave Base Address Table

The resources that appear as VME Slave devices in the Model 4200/01A are the Inter-processor Communications Flag (ICF) Registers (in A16 space), and the SRAM and the MIX Bus, which can be mapped into either A32 or A24 space. The default addresses for these resources, shown at the bottom of the power-up logo display, are 0x0000 (A16) for the ICF Register, 0x0000 0000 (A32) for the SRAM, and 0x4000 0000 (A32) for the MIX bus.

If this addressing scheme creates a conflict with other instruments in your system, the addresses can be changed. When the 4200/01A is reset, a pair of jumper blocks (JB2 and JB3 on the VME board, see Section 2.2.2, above) to select a group of addresses from a table stored in the User/Configuration Flash EEPROM. The contents of this table can be modified in the Monitor program. Pressing eight **(to Select VME Base Address Settings)** and **<Enter>** when the Monitor's main menu is displayed will bring a copy of the table to your display, with the VME Base Address Menu, shown in Figure 2-5, below.

```
          Current 4200/4201A VME Base Address Settings

 JB3   JB2          SRAM                  MIX          MIX SIZE         ICF
 2-3   2-3   1) A32=0x00000000   2) A32=0x40000000    9) 1000   13) A16=0x0000
 2-3   1-2   3) A32=0x80000000   4) A32=0xc0000000   10) 1000   14) A16=0xa000
 1-2   2-3   5) A24=0x00000000   6) A24=0x00400000   11)    4   15) A16=0x8000
 1-2   1-2   7) A24=0x00800000   8) A24=0x00c00000   12)    4   16) A16=0xc000

Select Index to be Modified(1..16), Q to Quit, or P to Program>
```

Figure 2-5: 4200/01A VME Base Address Menu

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.4 Changing the VME Slave Base Address Table (continued)

The menu shown in Figure 2-5, on the previous page, lists the four possible settings of the jumper blocks, and the VME Slave address settings associated with each. If none of these selections will work out well in your system, the table selections can be changed by selecting the number of the resource index you wish to modify.

For example, let's assume you will be working with the 4200/01A in A32 space, and that you want the MIX bus access region to begin at 0xC000 0000, regardless of where the SRAM is mapped (i. e., you wish to be able to re-map SRAM by changing the JB3 setting as shown in the table above, but you want MIX to remain at the same address). In this case, you would select item **2** from the menu above, and enter **c0000000** at the **Select New Address** prompt, as shown in Figure 2-6, below.

```
Select Index to be Modified(1..16), Q to Quit, or P to Program>2
Current Address is 0x40000000, Select New Address >c0000000


          Current 4200/4201A VME Base Address Settings

 JB3   JB2          SRAM                MIX          MIX SIZE         ICF
 2-3   2-3   1) A32=0x00000000   2) A32=0xc0000000    9) 1000   13) A16=0x0000
 2-3   1-2   3) A32=0x80000000   4) A32=0xc0000000   10) 1000   14) A16=0xa000
 1-2   2-3   5) A24=0x00000000   6) A24=0x00400000   11)    4   15) A16=0x8000
 1-2   1-2   7) A24=0x00800000   8) A24=0x00c00000   12)    4   16) A16=0xc000


Select Index to be Modified(1..16), Q to Quit, or P to Program>
```

Figure 2-6: Changing a Base Address

In order for any changes made to the table to take effect, the change must be programmed into the Flash EEPROM and the 4200/01A must be reset. Type **P** and **<Enter>** at the Base Address Menu prompt to save the changes. Then press the front panel Reset button to load the new table. The resulting displays should appear as shown in Figure 2-7, on the next page.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.4 Changing the VME Slave Base Address Table (continued)

```
Select Index to be Modified(1..16), Q to Quit, or P to Program>p

Erasing Device...Done

Verifying Erase...Done


                  ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
                 ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
                ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
    ]]]]]]]]]  ]         ]]]    ]]]  ]]]          ]  ]]]]]]]]]   ]]     ]]
    ]]     ]]  ]  ]]]]]]]]]]  ]  ]]  ]]]]]]]  ]]]]   ]]          ]]   ]]
    ]]]]]]]]] ]]         ]]]  ]]  ]  ]]]]]]]  ]]]    ]]]]]]]     ]]]]
    ]]       ]]]  ]]]]]]]]]]  ]]]    ]]]]]]]  ]]     ]]          ]]   ]]
    ]]      ]]]]         ]]]  ]]]]   ]]]]]]]  ]      ]]]]]]]]]   ]]     ]]
           ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
          ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]  Copyright 1993 (c) Pentek, Inc.
         ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]  Model 4200/4201A - Revision 2.0




2 MByte Memory Installed
A32 Base Address for RAM = 00000000
A32 Base Address for MIX = c0000000
A16 Base Address for ICF = 0000
>
```

Figure 2-7: Saving New Addresses to Flash Memory

Other address changes can be made in a similar manner, by selecting the appropriate menu option (**1-8** or **13-16**). Note that JB3 selects the MIX/SRAM address space as either A32 or A24. The table CANNOT be modified to provide an A32 Address with a jumper on JB3 between pins 1 & 2. Nor can the table be modified to put SRAM in A32 space and MIX in A24, or vice-versa. Likewise, the ICF register must always appear in A16 space and cannot be mapped into A32 or A24 space.

The other menu mapping consideration that can be changed from within this menu is the amount of memory allotted to the MIX bus. By default, the MIX bus is given a 1 GB region if A32 space is selected by JB3, or a 4 MB region if A24 space is selected. These regions can be decreased by a factor of two or four, or MIX access can be disabled by setting the size of this region to 0. An example is shown In Figure 2-8, on the next page.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.4 Changing the VME Slave Base Address Table (continued)

```
          Current 4200/4201A VME Base Address Settings

 JB3   JB2          SRAM                MIX          MIX SIZE          ICF
 2-3   2-3   1) A32=0x00000000   2) A32=0xc0000000    9) 1000   13) A16=0x0000
 2-3   1-2   3) A32=0x80000000   4) A32=0xc0000000   10) 1000   14) A16=0xa000
 1-2   2-3   5) A24=0x00000000   6) A24=0x00400000   11)    4   15) A16=0x8000
 1-2   1-2   7) A24=0x00800000   8) A24=0x00c00000   12)    4   16) A16=0xc000

Select Index to be Modified(1..16), Q to Quit, or P to Program>9
Current Mix Module Size = 1000 MBytes, Select New Size(0, 256, 512, 1000)>256
          Current 4200/4201A VME Base Address Settings

 JB3   JB2          SRAM                MIX          MIX SIZE          ICF
 2-3   2-3   1) A32=0x00000000   2) A32=0xc0000000    9)  256   13) A16=0x0000
 2-3   1-2   3) A32=0x80000000   4) A32=0xc0000000   10) 1000   14) A16=0xa000
 1-2   2-3   5) A24=0x00000000   6) A24=0x00400000   11)    4   15) A16=0x8000
 1-2   1-2   7) A24=0x00800000   8) A24=0x00c00000   12)    4   16) A16=0xc000

Select Index to be Modified(1..16), Q to Quit, or P to Program>
```

Figure 2-8: Changing the MIX Allocation Region

Once again, any changes must be programmed into the Flash memory and the 4200/01A must be reset for the changes to become effective.

To leave this menu and return to the main Monitor menu, enter a **Q**. For other methods of changing the VME Slave Address Map, see Sections 2.4.14.1 and 2.4.14.2.

2.4.5 Changing the 68030's Memory Map

Selecting option **9 (to Select Internal Memory Map)** from the Monitor's main menu will bring you to a screen that displays the 4200/4201A Internal Memory Map. This is also the map of the Model 4200A's resources as seen by the 68030, and by an Upper MIX Bus Master. The terminal displays that will be seen during this procedure are shown in Figure 2-9, on the next page.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.5 Changing the 68030's Memory Map (continued)

```
Enter
 1 to Examine Memory
 2 to Upgrade Firmware
 3 to Download Program
 4 to Run Program
 5 to Stop Program
 6 to Load Program to User Flash EEProm
 7 to Copy Program from User Flash EEProm
 8 to Select VME Base Address Settings
 9 to Select Internal Memory Map
10 to Select VME Attributes
11 to Select Interrupt Settings
12 to Select Other Parameters
13 to Select Other Commands
>9

          Current 4200/4201A Internal Memory Map

 0x00000000 - 0x003fffff      4 MB SRAM
 0x00400000 - 0xbeffffff      A32/D32 VMEbus Master Access
 0xbf000000 - 0xbfffffff      D32 VSBbus Master Access
 0xc0000000 - 0xefffffff      Mix Module Access
 0xf2000000                   MIX Flag Register Access
 0xf4000000 - 0xf4ffffff      A24/D32 VMEbus Master Access
 0xf5000000 - 0xf5ffffff      A24/D16 VMEbus Master Access
 0xff000000 - 0xff03ffff      128 kB Flash Prom #1 (Monitor)
 0xff040000 - 0xff04ffff      128 kB Flash Prom #2 (Config)
 0xfff20000                   MIX Control Register Access
 0xfffe0000 - 0xfffeffff      A16/D32 VMEbus Master Access
 0xffff0000 - 0xffffffff      A16/D16 VMEbus Master Access

 Select 'V' to Modify VSBbus Master Access Region Size,
        'Q' to Return to Main Menu,
        'P' to Program Flash Memory
        Enter Selection>
```

Figure 2-9: The Monitor's 68030 Memory Map Display

The only parameter of the Master memory map that can be modified under this menu at the time of this writing is the size of the VSB Master Access region, which can be increased at the expense of the A32/D32 VME Master Access region located just before it. The default size of the VSB Master Access region is 16 MB.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.5 Changing the 68030's Memory Map (continued)

Suppose, for example, it was your intention to work with the 4200A mainly as an A24 VMEbus Master, and had little use for A32 VME Master access. In this case, you could devote a 1 GB region to A32 VSB Master transactions, by entering a **`V (to modify VSBus Master Access Region Size)`** at the **`Enter Selection`** prompt, and then entering **`1024`** in response to the **`Select New Address Size`** prompt. This example is illustrated in Figure 2-10, below.

```
Select 'V' to Modify VSBbus Master Access Region Size,
       'Q' to Return to Main Menu,
       'P' to Program Flash Memory
       Enter Selection>v
 Current VSB Address Size = 16 MBytes, Select New Address Size(16..3068)>1024

         Current 4200/4201A Internal Memory Map

 0x00000000 - 0x003fffff      4 MB SRAM
 0x00400000 - 0x7fffffff      A32/D32 VMEbus Master Access
 0x80000000 - 0xbfffffff      D32 VSBbus Master Access
 0xc0000000 - 0xefffffff      Mix Module Access
 0xf2000000                   MIX Flag Register Access
 0xf4000000 - 0xf4ffffff      A24/D32 VMEbus Master Access
 0xf5000000 - 0xf5ffffff      A24/D16 VMEbus Master Access
 0xff000000 - 0xff03ffff      128 kB Flash Prom #1 (Monitor)
 0xff040000 - 0xff04ffff      128 kB Flash Prom #2 (Config)
 0xfff20000                   MIX Control Register Access
 0xfffe0000 - 0xfffeffff      A16/D32 VMEbus Master Access
 0xffff0000 - 0xffffffff      A16/D16 VMEbus Master Access

 Select 'V' to Modify VSBbus Master Access Region Size,
       'Q' to Return to Main Menu,
       'P' to Program Flash Memory
       Enter Selection>
```

Figure 2-10: Changing the VSB Allocation Region

Note that the size of the VME A32/D32 Master region has been decreased to allow for the increased size of the VSB Master region. If desired, A32/D32 VME Master Access can be disabled entirely, and the full 3 GB region from 0x0040 0000 to 0xBFFF FFFF devoted to VSB Master Access, by entering **`3068`** at the **`Select New Address Size`** prompt.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.5 Changing the 68030's Memory Map (continued)

As was the case with changes to the VME Slave Address Table, changes to the 68030 Memory Map must be programmed into the Flash memory and the 4200/01A must be reset in order for the changes to take effect. To accomplish this, select **`P (to Program Flash Memory)`** from the **`Enter Selection`** prompt, and then press the front Panel Reset button.

2.4.6 VME Attributes

The selections under the VME Attributes Menu in the Monitor program allow the user to select the VMEbus Request Level and Arbitration method. Selecting option **`10 (to Select VME Attributes)`** from the Monitor's main menu will bring the VME Attributes menu to the terminal screen. An example is shown in Figure 2-11, below.

```
>
Enter
 1 to Examine Memory
 2 to Upgrade Firmware
 3 to Download Program
 4 to Run Program
 5 to Stop Program
 6 to Load Program to User Flash EEProm
 7 to Copy Program from User Flash EEProm
 8 to Select VME Base Address Settings
 9 to Select Internal Memory Map
10 to Select VME Attributes
11 to Select Interrupt Settings
12 to Select Other Parameters
13 to Select Other Commands
>10

          Current 4200/4201A VME Attributes

 1) Bus Request Level  - 3
 2) Arbitration Method - Round Robin

Select Index to be Modified(1..2), Q to Quit, or P to Program>
```

Figure 2-11: The Monitor's VME Attributes Menu

To change the VMEbus Request Level, select option **`1 (Bus Request Level)`** from the menu above. An example is presented on the next page, in Figure 2-12.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.6 VME Attributes (continued)

```
          Current 4200/4201A VME Attributes

 1) Bus Request Level  - 3
 2) Arbitration Method - Round Robin

Select Index to be Modified(1..2), Q to Quit, or P to Program>1
Current Bus Request Level is 3, Select New Level >2

          Current 4200/4201A VME Attributes

 1) Bus Request Level  - 2
 2) Arbitration Method - Round Robin

Select Index to be Modified(1..2), Q to Quit, or P to Program>
```

Figure 2-12: Changing the 4200/01A's Bus Request Level

To change the VMEbus Arbitration method, select option **2 (Arbitration Method)** from the VME Attributes menu. Figure 2-13, below, presents an example of this procedure.

```
    Current 4200/4201A VME Attributes

 1) Bus Request Level  - 2
 2) Arbitration Method - Round Robin

Select Index to be Modified(1..2), Q to Quit, or P to Program>2
Current Arbitration Method is Round Robin,
Select New Method(0 - Round Robin, 1 - Priority) >1

          Current 4200/4201A VME Attributes

 1) Bus Request Level  - 2
 2) Arbitration Method - Priority

Select Index to be Modified(1..2), Q to Quit, or P to Program>
```

Figure 2-13: Changing the 4200/01A's Arbitration Method

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.6 VME Attributes (continued)

As was the case with changes to the VME Slave Address Table and the 68030 Memory Map, VME Attribute changes must be programmed into the Flash memory and the 4200/01A must be reset in order for the changes to take effect. To accomplish this, select **`P (to Program)`** from the **`Select Index`** prompt, and then press the front Panel Reset button.

2.4.7 Examining and Modifying 4200/01A Memory Contents

The 4200/01A Monitor provides a feature that permits the user to select a memory location, examine its contents, and modify those contents, if necessary. This part of the program is started by selecting option **`1 (to Examine Memory)`** from the main Monitor prompt. This will generate a new prompt, **`Enter 68030 Address (. to quit):`**. Enter any valid address in response to this prompt, and the terminal will list the address and its data, as shown in Figure 2-14, below.

```
Enter
 1 to Examine Memory
 2 to Upgrade Firmware
 3 to Download Program
 4 to Run Program
 5 to Stop Program
 6 to Load Program to User Flash EEProm
 7 to Copy Program from User Flash EEProm
 8 to Select VME Base Address Settings
 9 to Select Internal Memory Map
10 to Select VME Attributes
11 to Select Interrupt Settings
12 to Select Other Parameters
13 to Select Other Commands
>1

Enter 68030 Address (. to quit): f000
0000f000 d7fbffff:
```

Figure 2-14: Examining a 4200/01A Memory Location

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.7 Examining and Modifying 4200/01A Memory Contents (continued)

Other keys that can be used in this Monitor utility are listed in Table 2-8, below:

Table 2-8
Pentek Model 4200/01A Monitor Program
Memory Examine/Modify Keystrokes

<Enter>	display contents of next longword address
-	display contents of previous longword address
=	re-examine or overwrite current longword address
.	quit to Monitor main menu

If, while the content of a given memory location is displayed, you type in a hex number and hit **=**, the number you typed will be written to that memory location, and the Monitor will re-display the contents of that location. If you simply hit the **=** key without first typing in a number, the location will be read again and the result of the new read will be displayed. This can be useful when polling a given location to see, for example, if a given operation has been completed.

If you wish to examine an address that is physically far removed from the range of addresses you are currently examining, exit to the Monitor's main menu by typing a **.** , and then enter a **1** at the **>** prompt to return to the Examine/Modify utility and enter the new address.

2.4.8 Loading Programs to the 4200/01A with the Monitor

User programs may be downloaded from a host computer to either of two destinations in the 4200/01A, specifically the SRAM or the Configuration/ User Flash EEPROM. The sections below describe these procedures.

2.4.8.1 Loading Programs to the 4200/01A's SRAM

Selecting option **3 (to Download Program)** from the main menu's **>** prompt will begin the SRAM download routine. The terminal will respond with the prompt **Ready for Program Download, Send File**. Programs loaded to the 4200/01A must be compiled in 68030-compatible Motorola Hex ASCII format, and transferred using ASCII protocols. See the documentation provided with your terminal emulation software for information on setting up an ASCII protocol transfer.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.8 Loading Programs to the 4200/01 with the Monitor (continued)

2.4.8.1 Loading Programs to the 4200/01A's SRAM (continued)

When compiling code to download to the 4200/01A, set up your command linker file such that the code will be downloaded to an SRAM address offset somewhere above 0x0001 0000 (the boot code is copied from EEPROM to SRAM in the area below 0x0001 0000). Refer to Section 3.1.4 of this manual for further information about limitations on SRAM usage. After the file transfer is complete, the terminal will display the message **`Program size is <your_code_size> bytes`** (the size will be given in decimal, not hex), and then return to the **>** prompt.

2.4.8.2 Loading Programs to the User/Configuration Flash EEPROM

Selecting option **`6 (to Load Program to User Flash EEProm)`** from the main menu's **>** prompt will start the EEPROM download routine. The terminal will respond to this entry with the prompt **`Enter Starting Address and Size of Program or 'Q' to quit>`**. As stated above, programs loaded to the 4200/01A must be compiled in 68030-compatible Motorola Hex ASCII format, and transferred using ASCII protocols. The documentation provided with your terminal emulation software will give information on setting up ASCII protocol transfers.

Code residing in EEPROM will be copied into SRAM before execution. When compiling the code you wish to load into the 4200/01A's User/Configuration Flash EEPROM you should set up the command linker file as if the code were being loaded to SRAM (see Section 2.4.8.1, above, and Section 3.1.4 of this manual for information about regions of SRAM where code should NOT be loaded).

In response to the prompt for the address and size of your code, enter the target SRAM address and the file size, in bytes, expressed as a decimal number (i. e., not in hex). The address and size should be separated by a space. The Monitor will then dump your program into SRAM, erase the existing Flash EEPROM code, verify that the PROM has been erased, burn your code into the Flash Memory, and, upon completion, will display the main menu's **>** prompt. An example is presented on the next page, in Figure 2-15.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.8 Loading Programs to the 4200/01A with the Monitor (continued)

2.4.8.2 Loading Programs to the User/Configuration Flash EEPROM (continued)

```
Enter
 1 to Examine Memory
 2 to Upgrade Firmware
 3 to Download Program
 4 to Run Program
 5 to Stop Program
 6 to Load Program to User Flash EEProm
 7 to Copy Program from User Flash EEProm
 8 to Select VME Base Address Settings
 9 to Select Internal Memory Map
10 to Select VME Attributes
11 to Select Interrupt Settings
12 to Select Other Parameters
13 to Select Other Commands
>6
Enter Starting Address and Size of Program or 'Q' to Quit > 50008 19724

Erasing Device...Done

Verifying Erase...Done

Loading Program into Flash Memory...Done
>
```

Figure 2-15: Saving a Program to Flash Memory

The code copied into the User/ Configuration Flash EEPROM in this manner will begin at address 0xFF04 0020. The first Flash Memory location (0xFF04 0000) will contain a code indicating that the factory supplied code has been erased (i. e., that the PROM now contains user code). The next longword location in the EEPROM (address 0xFF04 0004) will contain the code's starting address, and the next location (0xFF04 0008) contains the file size. Both of these parameters are expressed as 32-bit hex numbers.

2.4.8.3 Copying EEPROM Code into SRAM

Code that resides in the Model 4200/01A's User/Configuration Flash EEPROM can be copied into SRAM for execution or editing by selecting option **`7 (to copy Program from User Flash EEProm)`** from the Monitor's main menu **`>`** prompt. an example of this procedure is shown on the following page, in Figure 2-16.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.8 Loading Programs to the 4200/01A with the Monitor (continued)

2.4.8.3 Copying EEPROM Code into SRAM (continued)

```
Enter
 1 to Examine Memory
 2 to Upgrade Firmware
 3 to Download Program
 4 to Run Program
 5 to Stop Program
 6 to Load Program to User Flash EEProm
 7 to Copy Program from User Flash EEProm
 8 to Select VME Base Address Settings
 9 to Select Internal Memory Map
10 to Select VME Attributes
11 to Select Interrupt Settings
12 to Select Other Parameters
13 to Select Other Commands
>7
User Code copied to SRAM
>
```

Figure 2-16: Copying a Program from EEPROM to SRAM

2.4.9 Starting and Stopping Programs from the Monitor

68030 programs that have been downloaded to the 4200/01A as described above can be started from the Monitor by selecting option **4 (to Run Program)** from main menu's **>** prompt. This action will generate a new prompt, **Enter Starting Address or 'Q' to Quit >**. Key in the starting address of the code you wish to run and hit **<Enter>**. The screen will then display the message **Executing Code at 0x<your_code_address>**, and the program will run.

A program that is running on the 68030 can be stopped from the Monitor by selecting option **5 (to Stop Program)** from the main menu's **>** prompt.

2.4.10 Upgrading 4200/01A Firmware

As we at Pentek receive more user feedback on the applications of this device, we may, from time to time, release new firmware with improved or updated features to our community of users. To simplify such changes, the 4200/01A's Monitor includes a Firmware Upgrade option, which is accessed by selecting option **2 (to Upgrade Firmware)** from the Monitor's main menu **>** prompt.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.10 Upgrading 4200/01A Firmware (continued)

Before proceeding with the steps below, make sure that BOTH of the Flash EEPROMs are write enabled. See Section 2.2.4 of this manual for details.

When you select this option, the Monitor will respond with a new prompt, **`Ready for Program Download, Send File.`** Files to be downloaded to the 4200/01A will be in Hex ASCII format, and sent using ASCII file transfer protocols. See the documentation provided with your terminal emulation software for information on setting up an ASCII protocol transfer.

The new firmware will initially be downloaded into SRAM. After the download is complete, the information and instructions shown below, in Figure 2-17, will be displayed.

```
Program size is <file_size> bytes
Press G to LOAD program into Flash Memory
Press A to ABORT
>
```

Figure 2-17: Monitor Display After New Firmware Download

To save this code into the Flash EEPROM, type a **`G`** at the **`>`** prompt. To quit without saving or running the new code, type an **`A`** at the **`>`** prompt. The character typed in response to this prompt will not echo. Entering **`A`** will return you to the Monitor's main menu **`>`** prompt. Entering **`G`** will cause the monitor to burn the new code into the 4200/01A's EEPROMs. Examples of the resulting displays are shown in Figure 2-18, on the next page.

Be aware that downloading new firmware to the 4200/01A will erase the unit's existing boot code. User-generated code should NOT be sent to the 4200/01A in this manner. For information about the proper methods for downloading user code, see Section 2.4.8.2.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.10 Upgrading 4200/01 Firmware (continued)

```
Enter
 1 to Examine Memory
 2 to Upgrade Firmware
 3 to Download Program
 4 to Run Program
 5 to Stop Program
 6 to Load Program to User Flash EEProm
 7 to Copy Program from User Flash EEProm
 8 to Select VME Base Address Settings
 9 to Select Internal Memory Map
10 to Select VME Attributes
11 to Select Interrupt Settings
12 to Select Other Parameters
13 to Select Other Commands
>2
Ready for Program Download, Send File
Program size is 42322 bytes
Press G to LOAD program into Flash Memory
Press A to ABORT

Erasing Device...Done

Verifying Erase...Done

Now programming rom...Done

Verifying ...Done


                  ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
                 ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
                ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
    ]]]]]]]]]  ]         ]]]    ]]]  ]]]          ]  ]]]]]]]]]   ]]     ]]
    ]]     ]]  ]  ]]]]]]]]]]  ]  ]]  ]]]]]]]  ]]]]   ]]          ]]   ]]
    ]]]]]]]]] ]]         ]]]  ]]  ]  ]]]]]]]  ]]]    ]]]]]]]     ]]]]
    ]]       ]]]  ]]]]]]]]]]  ]]]    ]]]]]]]  ]]     ]]          ]]   ]]
    ]]      ]]]]         ]]]  ]]]]   ]]]]]]]  ]      ]]]]]]]]]   ]]     ]]
           ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
          ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]  Copyright 1993 (c) Pentek, Inc.
         ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]  Model 4200/4201A - Revision 2.0


2 MByte Memory Installed
A32 Base Address for RAM = 00000000
A32 Base Address for MIX = 40000000
A16 Base Address for ICF = 0000
>
```

Figure 2-18: Monitor Displays for Burning New Firmware into Flash Memory

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.11 Selecting Interrupt Settings with the Monitor

This feature was not yet implemented at the time of this release of 4200/01A firmware. Interested parties may contact Pentek at the phone number shown on the cover page of this manual for information about projected dates for subsequent firmware releases.

2.4.12 Automatically Executing User Code

Only one option exists under the Monitor's main menu selection for "Other Parameters" at the time of this release of 4200/01A firmware, and that is to configure the unit to automatically run user code after a reset or power-up. This option is accessed by selecting **`12 (to Select Other Parameters)`** from the Monitor's main menu **`>`** prompt. As with the features described above, a change in the state of this option must be saved into Flash Memory.

Figure 2-19, on the next page, shows the monitor displays resulting from the selection of this option when the function of the user program that is loaded is to print **`Hello World`** to the terminal display.

2.4.13 Issuing Other Commands with the Monitor

This feature was not yet implemented at the time of this release of 4200/01A firmware. Interested parties may contact Pentek at the phone number shown on the cover page of this manual for information about projected dates for subsequent firmware releases.

2.4.14 VMEbus Access to Monitor Features

All of the setup and programming utilities provided by the Monitor Program are also available to VMEbus Masters. The sections below describe the methods that are used to access these functions over the VMEbus. These changes can also be made by manually re-programming the Flash

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

EEPROMs. That topic is covered in Section 3.3 of this manual.

```
Enter
 1 to Examine Memory
 2 to Upgrade Firmware
 3 to Download Program
 4 to Run Program
 5 to Stop Program
 6 to Load Program to User Flash EEProm
 7 to Copy Program from User Flash EEProm
 8 to Select VME Base Address Settings
 9 to Select Internal Memory Map
10 to Select VME Attributes
11 to Select Interrupt Settings
12 to Select Other Parameters
13 to Select Other Commands
>12

          Current 4200/4201A Parameters

 1) Execute User Code after Reset - OFF

Select Index to be Modified(1), Q to Quit, or P to Program>1
Current Execute User Code Selection is OFF,
Select New Choice(0 - Off, 1 - On) >1

          Current 4200/4201A Parameters

 1) Execute User Code after Reset - ON

Select Index to be Modified(1), Q to Quit, or P to Program>p

Erasing Device...Done

Verifying Erase...Done

Loading Program Into Flash Memory...Done


                  ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
                 ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
                ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
    ]]]]]]]]]  ]         ]]]    ]]]  ]]]         ]  ]]]]]]]]]   ]]     ]]
    ]]     ]]  ]  ]]]]]]]]]]  ]  ]]  ]]]]]]] ]]]]   ]]          ]]   ]]
    ]]]]]]]]] ]]         ]]]  ]]  ]  ]]]]]]] ]]]    ]]]]]]]     ]]]]
    ]]       ]]]  ]]]]]]]]]]  ]]]    ]]]]]]] ]]     ]]          ]]   ]]
    ]]      ]]]]         ]]]  ]]]]   ]]]]]]] ]      ]]]]]]]]]   ]]     ]]
           ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
          ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]  Copyright 1993 (c) Pentek, Inc.
         ]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]  Model 4200/4201A - Revision 2.0



2 MByte Memory Installed
A32 Base Address for RAM = 00000000
A32 Base Address for MIX = 40000000
A16 Base Address for ICF = 0000
>User Code loaded into SRAM
Executing Code at 0x50008
Hello World
Hello World
```

Figure 2-19: Executing User Code After Reset

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.14 VMEbus Access to Monitor Features (continued)

2.4.14.1 Host Interface Program - 4200.C

Included among the Programming Examples on the enclosed diskette (described in Appendix B of this manual) is a program called 4200.C. This program's purpose is to provide an alternate user interface for the functions provided by the Monitor in the absence of an RS-232 terminal. All of the displays and data entry occur on the host computer's console. A "host", in this sense, is a computer system that has Master access to the VMEbus on which the 4200/01A resides, thus having the capability to read and write the 4200/01A's SRAM. If the use of an RS-232 terminal is problematic within the framework of your application, we strongly recommend that you compile this sample code on your host system, and use it to configure the 4200/01A.

The functions performed by this program are invoked by commands, and associated parameters, deposited in the 4200/01A's SRAM. The commands (and the parameters, if necessary) are placed in pre-defined locations in SRAM, and then the command location is monitored by the host for a ready indication returned by the 68030. The 68030 will write a '0' to the command code location to indicate successful completion of the command, a '-1' to indicate that an error occurred during execution of the command, or a '-2' to indicate that the requested command is not available. The parameters, if any are required, should be planted at their proper locations **BEFORE** the command code is written.

Table 2-9, below, gives the SRAM locations where the command codes and parameters must be placed. Note that the generic DMA routine uses different locations for the parameters than any of the other commands. Table 2-10, on the next page, lists the available functions, their command codes and the applicable parameters.

Table 2-9: 4200.C Host Interface Program - SRAM Useage

At SRAM Location...	We place the...
SRAM_base+0x00	Command Code
SRAM_base+0x04	Parameter 1
SRAM_base+0x08	Parameter 2
SRAM_base+0x44	DMA Parameter 1
SRAM_base+0x48	DMA Parameter 2
SRAM_base+0x4C	DMA Parameter 3
SRAM_base+0x50	DMA Parameter 4

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.14 VMEbus Access to Monitor Features (continued)

2.4.14.1 Host Interface Program - 4200.C (continued)

Table 2-10
Pentek Model 4200/01A
Host Interface Program - Functions, Codes and Parameters

Function	Cmd. Code	Parameter 1	Parameter 2	Parameter 3	Parameter 4
Read Data (Look)	0x01	68030 Address	N/A	N/A	N/A
Write Data (Set)	0x02	68030 Address	Hex Data	N/A	N/A
Run Code at 0x0005 0008	0x03	N/A	N/A	N/A	N/A
Set A32 Table	0x05	Base Address	Table Index	N/A	N/A
Set A24 Table	0x06	Base Address	Table Index	N/A	N/A
Set A16 Table	0x07	Base Address	Table Index	N/A	N/A
Run Code at Address	0x08	68030 Address	N/A	N/A	N/A
Load User Code to Flash Memory	0x0A	68030 Start Address	Code Size (bytes)	N/A	N/A
Copy User Flash Code to SRAM	0x0B	N/A	N/A	N/A	N/A
*Generic DMA	0x20	68030 Src. Addr.	68030 Dest. Addr.	# of lw transfers	Control Word

* - The DMA routine uses different parameter locations than the commands above. See Table 2-9.

The generic DMA routine referred to in the table above can accomodate block transfers of up to 65536 32-bit words in length (i. e., parameter 3 for the DMA command must be a number between 1 and 65536). The Control Word (parameter 4) is 5-bits long. the bits in the control word are defined below.

Bit 0 - When set to '1', enables increment of source address.
Bit 1 - When set to '1', enables increment of destination address.
Bit 2 - When set to '1', enables generation of VME IRQ when DMA completes.
Bit 3 -
Bit 4 - } binary-coded IRQ1-7 (bit 3 is LSB), vectors = 0xC1 - 0xC7
Bit 5 -

It should also be noted that the table above contains two different commands to execute user code that has been downloaded to the Model 4200/01A. Command code 0x03 requires no parameters, and will immediately begin execution of the code at code at address 0x0005 0008. Command code 0x08 (listed in the same table) requires one parameter, and that is the address at which execution should begin. This command also results in immediate execution. For both commands, control is passed via a Branch to Subroutine call (BSR). This structure allows a simple RTS call to return control to the the 4200.C program.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.14 VMEbus Access to Monitor Features (continued)

2.4.14.2 Host Interface Program Interrupt Access

Most of the commands provided by the host monitor can also be accessed by sending an interrupt to the 68030. If you write a '1' to address A16_base+0x22, and follow it by writing a '1' to A16_base+0x23, an interrupt is sent to the 68030 that causes it to run from the top of the EEPROM. This will halt the execution of any properly executing program and return the monitor to the command mode.

The reset mechanism described above will NOT help in the case of massive memory overwriting or other catastrophic situations.

In a manner similar to the method described in the previous section, used for accessing monitor-type commands from the host, command codes and parameters are deposited in specified SRAM locations. In this case, it does not matter whether the commands or the parameters are planted first, because command execution is commenced by generating an interrupt to the 68030. The interrupt is generated by writing a '1' to address A16_base+0x20, and then writing another '1' to A16_base+0x21. As before, the 68030 overwrites the command location to indicate completion of the command. A '0' in the command code location indicates successful completion of the command, a '-1' indicates that an error occurred during execution of the command, and a '-2' indicates that the requested command is not available.

Table 2-11
4200.C Host Interface Program Interrupt Access SRAM Useage

At SRAM Location...	We place the...
SRAM_base+0x40	Command Code
SRAM_base+0x44	Parameter 1
SRAM_base+0x48	Parameter 2

Table 2-12
Pentek Model 4200/01A Host Interface Program Interrupt Access - Functions, Codes and Parameters

Function	Cmd. Code	Parameter 1	Parameter 2
Read Data (Look)	0x28	68030 Address	N/A
Write Data (Set)	0x29	68030 Address	Hex Data
Run Code at Address	0x2A	68030 Address	N/A
Set A32 Table	0x2C	Base Address	Table Index
Set A24 Table	0x2D	Base Address	Table Index
Set A16 Table	0x2E	Base Address	Table Index
Stop Program	0x2F	N/A	N/A
Load User Code to Flash Memory	0x30	68030 Start Address	Code Size (bytes)
Copy User Flash Code to SRAM	0x31	N/A	N/A

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.14 VMEbus Access to Monitor Features (continued)

2.4.14.2 Host Interface Program Interrupt Access (continued)

Table 2-11, at the bottom left of the previous page, gives the SRAM locations to be used for placement of the commands and their associated parameters, if any. Table 2-12 (previous page, bottom right) lists the available functions, their command codes and the applicable parameters.

Like Table 2-10, in the previous subsection, this table also contains a command to execute user code that has been downloaded to the Model 4200/01A. Command code 0x2A requires the starting address of the code for its sole parameter. Unlike the commands in Table 2-10, however, when this command is issued, execution does not begin until the 68030 receives an interrupt, as is the case for all the commands in Table 2-12. This command also passes control to the user code via a Branch to Subroutine call (BSR). Again, an RTS call will return control to 4200.C.

2.4.14.3 Modifying A32 or A24 Base Addresses from VME A16 Space

Another method for changing the VME Slave memory map of the 4200/01A without using the Monitor program involves the assumption that one of the four default A16 base addresses will be acceptable in your system. This scheme will allow the SRAM and the MIX bus to be placed at user-defined locations in A24 or A32 space. The downside of this method is that the new addresses are NOT saved in the non-volatile Flash EEPROM. Thus, if this method is chosen, the procedure must be repeated whenever the board is reset or power is cycled.

In this method, the desired address is re-constructed from a series of three bytes (for A24 access) or four bytes (for A32 access) written to the A16 locations indicated in Table 2-13, at the top left of the next page. These addresses differ, depending upon the type of host processor in use. The even locations indicated in the table assume the use of an Intel (i. e., PC-type) host. The odd locations, listed in parentheses, would be used in Motorola-type systems (e. g., a SUN Workstation). Another byte is then written to the A16 base address to define what the given address will be used for, as shown in Table 2-14 (next page, top right).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.4 The 4200/01A Monitor Program (continued)

2.4.14 VMEbus Access to Monitor Features (continued)

2.4.14.3 Modifying A32 or A24 Base Addresses from VME A16 Space (continued)

Table 2-13
Pentek Model 4200/01
Byte Positions For New SRAM/MIX Base Address in A16 Space

A16 Address	Byte of SRAM/MIX Base Address
A16_Base+8(9)	MSB of A32 SRAM/MIX Base Address
A16_Base+6(7)	MSB of A24 SRAM/MIX Base Address or Byte 3 of A32 SRAM/MIX Base Address
A16_Base+4(5)	Byte 2 of A32/A24 SRAM/MIX Base Address
A16_Base+2(3)	LSB of A32/A24 SRAM/MIX Base Address

Table 2-14

To use the data at the A16 locations listed at the left as the...	Write the data below to A16_Base
A32 SRAM Base Address	1
A32 MIX Base Address	2
A24 SRAM Base Address	3
A24 MIX Base Address	4

2.4.14.4 Changing the VSB Access Region Size from the VMEbus

At the time of this writing, the only modification that can be made to the 68030's memory map is changing the size the 4200A's VSB access region, by decreasing or increasing the A32/D32 VME Master access region. To accomplish this end over the VMEbus, simply write the upper 16 bits of the desired VSB Base Address as an unsigned word to 68030 address 0xFFFD 0600, which is a boundary address register in the VAC. The data written to this location must be greater than or equal to 0x0040 (0x0040 0000 is the beginning of the A32/D32 VME Master access region), and less than or equal to 0xBFFF. This change is not saved to the Flash EEPROM, so a change made in this manner will be lost when power is cycled or a Global Reset is issued.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.5 Removing and Installing the Model 4200/01A Front Panel

In the natural course of working with card-cage based instrumentation, VME or otherwise, one often finds that he has more cards than he has slots in the cage. Should you be faced with such a problem, you may find it convenient to utilize a MIX stack assembly technique known as nested installation. When this technique is used, the baseboard and the first expansion module share a single slot in the VME cage, leaving a slot free for another instrument. In this nested assembly scheme, the Baseboard's front panel is removed, and the baseboard "shares" the first expansion module's front panel. Instructions for removing the 4200/01A's front panel (and for reinstalling it, when the need arises) can be found below. Figures 2-1 and 2-2, at the beginning of this chapter, show the three boards used in the 4200/01A and the mounting holes referred to in the procedures described below. For further information about nested MIX stack assembly, see Appendix C.

CAUTION!!

Perform all the assembly steps described on the pages that follow in a static controlled work environment.

2.5.1 Tools Required

1) 3/16" Slotted-Head Screwdriver
2) #1 Philips-Head Screwdriver
3) DIP-16 IC Extraction Tool (optional)

2.5.2 Removing the 4200/01A Mezzanine Board

The cable that connects the Model 4200/01A's Front Panel DB-25 connectors to the Serial Port electronics is attached by means of a 16-pin Dual Inline header. The header plugs into a socket which had to be located underneath the mezzanine board, to allow connection to the ports without interference with a nested expansion module. Thus, the mezzanine board must be removed before the front panel can be either removed or installed.

Place the assembled 4200/01A on your work surface with the mezzanine board facing up, and the front panel toward you. Although there are components located on both sides of the 4200A's VME board, we refer to the side that is now face up as the component side, and the side that is now face down as the solder side.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.5 Removing and Installing the Model 4200/01A Front Panel (continued).

2.5.2 Removing the 4200/01 Mezzanine Board (continued)

The mezzanine board is secured to the VME board by means of four nylon screws, at the corners of the mezzanine board. Remove these with the 3/16" slotted screwdriver and set them aside for use when reinstalling the mezzanine board.

The electrical connections between the 68030 and SRAM on the mezzanine board and the VIC/VAC chip set and its support components on the VME board are accomplished by means of a pair of shrouded, microminiature pin and socket connectors. The shrouded male connectors are located on the mezzanine board, and the female connectors are on the VME board. An 80-pin connector is located at the left front of the mezzanine board, and a 50-pin connector is at the mezzanine board's right rear. Pry the boards apart gently, taking care not to apply excessive torque to the pins. Set the mezzanine board aside in a safe place for reinstallation after the panel has been removed (or installed).

2.5.3 Removing or Installing the Front Panel Board Header

All electrical connections between the Model 4200/01A's VME board and Front Panel board are made via a 16-pin dual inline header. This header plugs into a set of low-profile insertion pins located on the VME board, labeled J3 (the label cannot be seen when the header is installed), just to the right of the 80-pin mezzanine board socket described in the section above.

If you are removing the panel, the header can now be removed with the DIP-16 extraction tool, or pried out with the slotted-head screwdriver. Alternatively, you may try to GENTLY pull the header out by the cable, but we DO NOT RECOMMEND THIS, as it stresses the connections within the header.

If you are installing the panel, the header is inserted, in the location described in the preceding paragraph, after the panel has been physically secured to the board. If the panel has been properly mounted, it is quite impossible to insert the header in the wrong orientation. However, we will note that the pin labeled '1' on the header is inserted in the socket pin pointed to by the small white arrowhead at the edge of the box drawn around the J3 connector pins on the card.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.5 Removing and Installing the Model 4200/01A Front Panel (continued)

2.5.4 Removing the Front Panel

The Model 4200/4201A's Front Panel is secured to the VME board by means of three screws, located on the board's solder side. As noted above, the name "solder side" is somewhat ambiguous, considering that there are components on both sides of the card. The solder side is the one on which the pins from the VME connectors are soldered, i.e., the side that has been described as "face down" until this point. So, to begin, turn the board over.

At the front, and roughly in the center of the VME card, is a single Philips-head screw, threaded into a mounting block on the front panel board. Remove this screw with the #1 Philips-head screwdriver, and set it aside momentarily.

There are two slotted-head screws at the extreme left and right front of the card. These are threaded into hex nuts that are part of the card ejector assemblies. Remove these screws with the 3/16" slotted-head screwdriver, holding a finger under the hex nut to hold it in place while turning the screw.

The panel may now be removed. We recommend that you take the time now to thread the three screws that you just removed from the VME board back into their former locations on the panel assembly, so they'll be readily available if (when) the panel is reinstalled. You may now go on to Section 2.5.6 for instructions on replacing the Mezzanine board.

2.5.5 Installing the Front Panel

The procedures listed above are ordered with the assumption that the first procedure performed will be removal of the panel, a natural enough assumption given that the unit is shipped with the panel installed. If you are installing the panel, rather than removing it, start by removing the Mezzanine board, as described in section 2.5.2, above.

Next, secure the panel assembly to the VME board, as follows. First, lay the panel assembly down on your work surface with the component side of the panel board face up, and the panel facing you. There should be a Philips-head screw threaded into a mounting block near the front center of the panel board. Remove this with the #1 Philips-head screwdriver. There should also be two slotted-head screws, passing through the card ejector assemblies at either end of the panel, threaded into hex nuts on the opposite sides of the assemblies. Remove these screws with the 3/16" slotted-head screwdriver, and set the screws and nuts aside momentarily.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

2.5 Removing and Installing the Model 4200/01A Front Panel (continued)

2.5.5 Installing the Front Panel (continued)

Next, orient the VME board with the VME connectors at the rear and the MIX connector (or the connector pattern, if no connector is installed at this time) to your left. With the panel assembly oriented as described in the preceding paragraph, place the VME board on top of the panel assembly, aligning the center mounting hole at the front of the VME board with the mounting block and the left and right mounting holes on the VME board with the holes in the ejector assemblies. Thread the Philips-head screw into the mounting block using the #1 Philips-Head screwdriver. Hold the hex nuts into their locations on the other sides of the ejector assemblies and thread the slotted-head screws into them with the 3/16" slotted-head screwdriver. The panel assembly is now secured to the VME board.

You may now turn the assembled unit over and install the Panel Board header into the J3 socket, as described in Section 2.5.3, above. Finally, replace the Mezzanine board, as described in Section 2.5.6, on the following page.

2.5.6 Replacing the Mezzanine Board

After removing or installing the front panel, the mezzanine board, which contains the 68030 processor and SRAM, must be replaced. To accomplish this, place the VME board, with or without the front panel assembly, on your work surface with the VME connectors facing up and to the rear. Hold the mezzanine board with the microminiature, shrouded pin connectors facing down, and the larger of these two connectors at the front and to the left. Align both sets of pin connectors with the mating socket connectors on the VME board GENTLY press down on the mezzanine board, in the areas just above the connector blocks, just until the connectors are fully seated. Secure the mezzanine board to the VME board with the four slotted-head nylon screws, using the 3/16" slotted head screwdriver. At this juncture, the installation or removal of the 4200/01A's front panel is complete and the unit is again ready to use.

2.6 Installing MIX Expansion Modules on the 4200/01A

For MIX module stack assembly procedures and other important information about Pentek's family of MIX expansion modules, please refer to Appendix C of this manual.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Chapter 3: Operation

3.1 Operating Theory

A detailed data flow diagram of the Pentek Model 4200/01A is presented in Figure 3-1, on the following page. Elements of this diagram will be referred to throughout the following discussion of the operating characteristics of the baseboard.

The essential function of the Model 4201A is to operate as a bridge between two bus systems, VME and MIX. The design goal was to provide a Master/Slave interface for each bus, allowing data transfer between them at a rate high enough to support real-time signal acquisition and processing in many applications. The Model 4200A expanded upon that theme by adding VSB Master interface capabilities.

This level of functionality was achieved by combining the VMEbus Interface Controller and VMEbus Address Controller (VIC/VAC) chip set from Cypress Semiconductor with a 32-bit 68EC030 processor from Motorola, operating at 40 MHz. The 68030 and the SRAM and Flash EEPROM reside on an intermediate or local bus. The local bus is generally under the control of the VIC, with assistance from some external arbitration and control logic.

The MIX bus and its resources appear as an extension of the local bus. When the 4200/01A assumes mastership of the MIX bus via this path, it is called a Lower MIX Bus Master (LMBM). This path also provides facilities that allow suitably equipped MIX modules to obtain mastership of the bus. An expansion module that has done this is known as an Upper MIX Bus Master (UMBM). The elements discussed above, along with some buffers that are controlled (directly or indirectly) by the VIC, are all there is to the Model 4201A.

The VSB capabilities in the Model 4200A are provided for by the addition of the VSB1400A/B chip set from PLX, with some additional buffering and multiplexing. This chip set was designed to give a 68030 the ability to become a VSB Master. Its implementation in the Model 4200A also allows an Upper MIX Bus Masters to become a VSB Master.

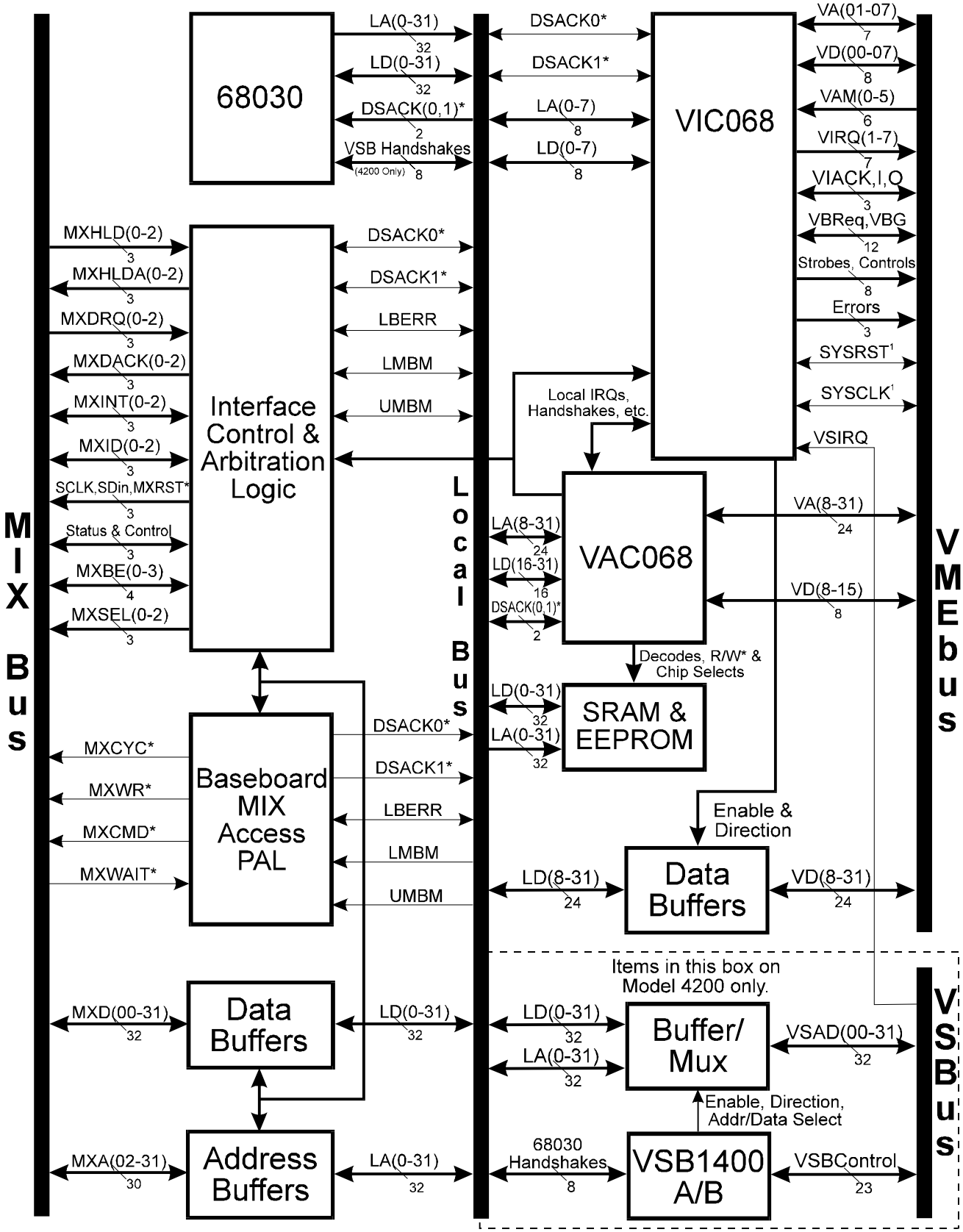
The major functional elements of the Model 4200/01A will be the topic of the sections that follow.

3.1.1 The VME Interface Controller (VIC068A)

The VIC068A was designed to directly provide complete VMEbus interface control and arbitration capabilities to Motorola's 68000 family of microprocessors. Specifically, the VIC directly connects to, and transacts with, the following VME signal lines: WRITE*, LWORD*, DS0*, DS1*, AS*, DTACK*, BERR*, BBSY*, BCLR*, ACFAIL*, SYSFAIL*, SYSRST*, SYSCLK*, IACK*, IACKIN*, IACKOUT*, IRQ(1-7)*, BR(0-3)*, BG(0-3)IN*, BG(0-3)OUT*, AM(0-5), A(01-07) and D(00-07). Additionally, control signals are provided for buffers to drive/receive the remaining address and data lines.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Figure 3-1: Pentek Model 4200/01A - Detailed Data Flow Diagram



1 - These lines are outputs if 4200/01 is configured as slot 1 controller, inputs otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.1 Operating Theory (continued)

3.1.1 The VME Interface Controller (continued)

Another of the VIC's jobs is the translation of local bus DSACK* signals to VMEbus DTACK* signals, and vice-versa, to denote the termination of a data cycle. It also serves a similar bidirectional translation role for the local bus PAS* and VMEbus AS* address strobe signals.

Master/Slave features supported by the VIC include read, write, write-posting and block transfer operations. VMEbus timing requirements are accommodated by a digital delay line, internal to the VIC, programmable in 7.8 nsec steps. The delay between local DSACK* signals and the VME DTACK* signal is also programmable. Timeout timers are provided for both local bus and VMEbus transactions.

The VIC allows for interleaved block transfers over the VMEbus, and (with some assistance from the VAC), acts as a DMA Master for transactions between the VMEbus and the 4200/01A's local bus, between VME and the Subsystem Bus (VSB), and between VME and the MIX bus. Burst count, transfer length, and interleaved period interval are all programmable, and local module-based DMA is also supported.

A total of 29 interrupts may be mapped through the VIC. It supports both interrupting and interrupt handling on all seven levels of the VMEbus. The VIC also accepts interrupts from the VSB, from the MIX bus and from the timers and serial port interface on the VAC, as Local IRQs. The VIC can generate local interrupts in response to VME interrupts, to the assertion of ACFAIL*, SYSFAIL*, or BERR*, to an arbitration timeout or to the completion of a local DMA transfer.

When the 4200/01A is configured as VME system controller and bus arbitrator, the VIC supports single level, prioritized, and round robin arbitration schemes. The fair request option is also supported by means of a fairness timer, programmable from 2 msec to 28 msec in 2 msec steps.

On the local bus side, the VIC can drive or receive the 8 least significant data and address lines (LD(0-7) and LA(0-7)). Other signals exchanged with the 68030 and the local bus arbitration logic (not all of these are shown in the flow diagram) are two size bits (which are used to determine whether a given data transaction involves byte, word, 3-byte or longword data elements), local bus error signals, the Read/Modify Control signal (RMC*, used to ensure the indivisibility of certain types of transactions), data and address strobes, the autovector signal, two Function Code signals, three interrupt priority level (IPL) signals, and reset and halt signals. The device most often in control of the local bus, and, by extension, of the MIX Bus, is the VIC.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.1 Operating Theory (continued)

3.1.1 The VME Interface Controller (continued)

Internal to the VIC, 58 programmable registers are used to allow for configuration control and status monitoring of VME and local operations. These registers, and their use, will be discussed in detail in Chapter 4.

3.1.2 The VME Address Controller (VAC068A)

The VAC068A augments the functionality of its companion part, the VIC, by adding complete, programmable decoding and mapping of the entire 16-, 24- and 32-bit VME address spaces. Separate segments, with programmable starting addresses and sizes, are available on the local bus side for VME Master and Slave accesses (including two independent slave base address registers), for the MIX bus and VSBus, and for the SRAM and the flash EEPROM.

The VAC's extended addressing capabilities also enhance the 4200A's block transfers, as the additional address counters allow for block sizes greater than 256 bytes. Other VAC functions include driving the Read/Write* signal for SRAM and Flash EEPROM cycles, and providing the DSACK* signal, to terminate data cycles involving SRAM and EEPROM access.

The dual, full duplex UART that drives the Model 4200A's Serial Ports is internal to the VAC. It also includes a programmable 16-bit timer, available for user-defined functions. Either of these internal peripherals is capable of generating interrupts, which are passed on to the VIC as Local IRQs.

3.1.3 The 68030

The 68030 can run programs stored in the Flash EEPROM or the SRAM. It can gain Master access to the VMEbus via the VIC, to the MIX bus via the VIC and the external support logic, and to the VSBus via the PLX chip set.

In the absence of an Upper MIX Bus Master, the 68030 is the only device that can master DMA transactions between the local bus and the MIX Bus, between the local bus and the VSBus, or between the MIX bus and the VSBus. The user should be aware that this is Software-controlled DMA, and as such is slower than the Hardware DMA implemented by the VIC/VAC chip set. These DMA transactions can be initiated by interrupt from VME, MIX, VSB, or the serial ports.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.1 Operating Theory (continued)

3.1.4 SRAM Allocation

The 4200/01A's shared SRAM is available for storage of user code and as a data exchange area. The user should be aware of two restrictions placed on the use of this memory, however. The first restriction applies to the lowest 40 kbytes (i. e., from SRAM_base to SRAM_base+0x0000 9FFF), where the boot code is copied from EEPROM at power-up.

The other portion of SRAM used by the system is the uppermost range of SRAM addresses, which is utilized as the stack. The stack begins at the highest address in SRAM (which will depend on the amount of memory your unit contains), and grows, as needed, down in addresses. Table 3-1, below, describes SRAM allocation. SRAM_top is equal to 0x000F FFFF for units equipped with 1MB of SRAM, 0x001F FFFF in units with 2 MB of SRAM, and 0x003F FFFF in units with 4 MB of SRAM.

Table 3-1
Pentek Model 4200/01A
SRAM Allocation

SRAM Region	Allocation
Bottom of Stack to SRAM_top	Stack (grows down as needed)
SRAM_base + 0x0000 A000 to Bottom Of Stack	Available RAM for User Code, etc.
SRAM_base to SRAM_base + 0x0000 9FFF	Boot Code

3.1.5 The MIX Bus Interface

As mentioned above, the MIX bus is treated as an extension of the 4200/01A's local bus. Access to MIX resources by the 68030 and by external VMEbus Masters is facilitated by a group of PALs that generate the MIX bus control and cycle termination signals. These types of MIX access are associated with the Lower MIX Bus Master (LMBM) condition.

When a MIX expansion module with bus master capabilities (e. g., a Model 4270, 4257 or 4254) arbitrates for and gains mastership of the MIX bus, that module also becomes master of the 4200/01A's local bus. This is the Upper MIX Bus Master (UMBM) condition.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.1 Operating Theory (continued)

3.1.6 4200/01A Data Cycles

An important key in understanding the mechanics of data transactions in the Model 4200/01A is in gaining an understanding of which parts initiate and terminate the various types of data cycles. In general, data cycles are initiated when the local bus master asserts a local Physical Address Strobe (PAS*). This signal is translated into a VME Address Strobe (AS*) by the VIC for transactions involving the VMEbus. During MIX bus cycles, the Baseboard MIX Access PAL reacts to a PAS* by asserting a MXCYC* signal.

The cycle termination begins with the assertion of a local bus DSACK* signal, usually asserted by the Slave or by a device acting as an agent for the Slave. For example, the VAC is responsible for asserting DSACK* for any transaction involving the SRAM or Flash EEPROMs, and the PLX VSB1400A/B chip set does the same during VSB cycles. The VIC translates the DSACK* signal into a DTACK* for VMEbus cycles. For MIX transactions, the Baseboard MIX Access PAL reacts to DSACK* by de-asserting the MXCYC* signal.

Table 3-2, below, lists the various types of data cycles that can occur on a 4200/01A-based system, and tells who is responsible for supplying the PAS* signal (i. e., who is the local bus master), and where the local DSACK* signal comes from.

Table 3-2
Pentek Model 4200/01A
Data Cycle Initiation/Termination

Type Of Data Cycle	Address Strobe Supplied by...	DSACK* Supplied by...
VME Slave Access to SRAM	VIC068	VAC068
VME Slave Access to MIX	VIC068	BB MIX Access PAL
VME Slave Access to ICF	N/A	VIC068[1]
68030 Access to SRAM	68030	VAC068
68030 Access to MIX	68030	BB MIX Access PAL
68030 Access to VIC Registers	68030	VIC068
68030 Master Access to VME	68030	VIC068
68030 Master Access to VSB	68030	VSB 1400A/B
VIC-Mastered DMA - VME/SRAM	VIC068	VAC068
VIC-Mastered DMA - VME/ MIX	VIC068	BB MIX Access PAL
VIC-Mastered DMA - VME/VSB	VIC068	VSB 1400A/B
UMBM Access to SRAM	UMBM	VAC068
UMBM Access to Another MIX Module	UMBM	N/A[2]

1 - No DSACK* necessary. VIC generates DTACK*
2 - Because MIX cycles are synchronous, they do not require a cycle termination. Slaves can hold off the end of a cycle by the properly timed assertion of MXWAIT*.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.1 Operating Theory (continued)

3.1.7 Local Bus Arbitration

On the following page is Figure 3-2, a block diagram showing the devices that can gain control of the local bus, and how they connect to the bus arbitration hardware. The critical function of a bus arbiter is to determine which device will be granted control of the bus in the event of either simultaneous requests, or a request from one device while another is in control.

The prioritization scheme used by the local bus arbiter is straightforward and strict, i. e., there is no "fair access" timer or rotating priority scheme associated with local bus mastership on the 4200/01A. Requests for the local bus from VME (via the VIC) are given the highest priority. The next highest priority request is from Upper MIX Bus Masters. Requests from MIX module 0 are given higher priority than requests from module 1, and module 1's priority exceeds that of module 2. Requests for local bus mastership from the 68030 receive the lowest priority. In fact, the 68030 cannot actively request the bus, but can take control if no other device has it.

If given device has been granted control of the local bus, and a bus request comes from a higher priority device, the current Bus Master is allowed to complete the transaction in progress before the arbiter forces it to relinquish control to the higher priority requester. Requests from lower priority devices are held off until the higher priority device de-asserts its bus request.

Once the 68030 has obtained the bus, however, it can set the Clear Bus bit in the MIX Control Register, (at 68030 address 0xFFF2 0000, see Section 3.2.3). The 68030 then retains local bus control until it chooses to clear that bit.

3.2 Model 4200/01A Memory Maps

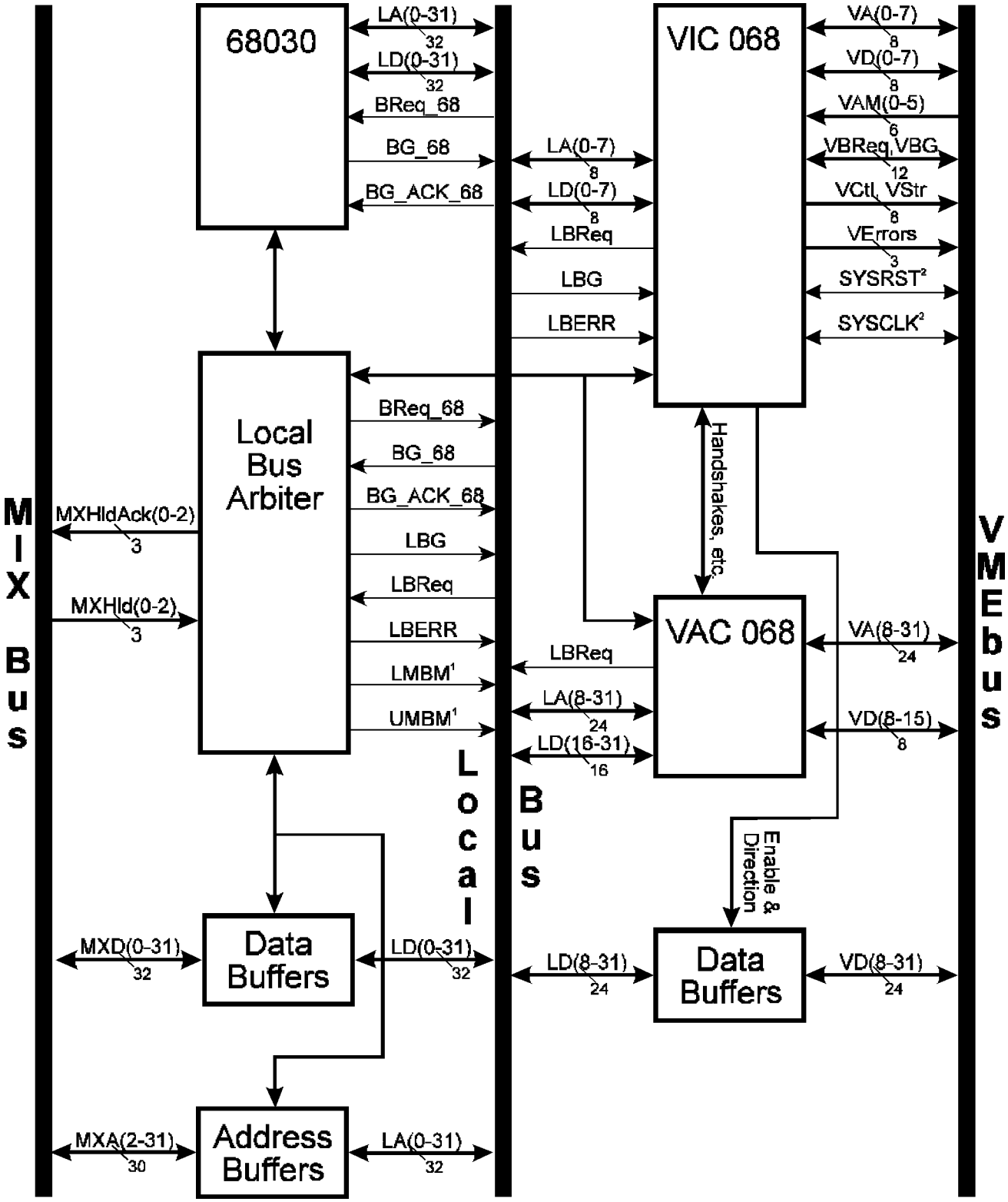
The sections that follow describe the default memory maps for the Model 4200/01A, from two distinct viewpoints; from the VMEbus and from the 68030. Also presented are maps of the registers associated with the MIX bus.

3.2.1 VME Slave Memory Map

The Model 4200/01A presents three resources to an external VMEbus Master when it is addressed as a Slave. These items are the VIC's seven Interprocessor Communications Flag (ICF) registers, which appear in A16 space, and the SRAM and the MIX bus, which may be mapped into A24 or A32 space. The ICF registers reside in the first 256 bytes of A16 space. The MIX bus access region in A32 space is 1 Gbyte (256 MByte/module, top 256 MB unused). In A24 space, the MIX module access region is 4 MByte (1MByte/module, top 1 MB unused). 4 MB of VME address space is assigned to the SRAM, in either A32 or A24 space, regardless of the actual amount of SRAM installed.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Figure 3-2: Pentek Model 4200/01A Local Bus Arbitration Block Diagram



¹LMBM = Lower MIX Bus Master = VIC when 4200/01 is a VME Slave, 68030 otherwise
UMBM = Upper MIX Bus Master = a MIX expansion module obtaining Bus Mastership
²These lines are outputs if 4200/01 is configured as slot 1 controller, inputs otherwise.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2 Model 4200/01A Memory Maps (continued)

3.2.1 VME Slave Memory Map (continued)

The base addresses for the slave resources listed in the previous paragraph are selected from a table in the User/Configuration Flash EEPROM. At power-up, and during Global Resets (see Section 2.3.4.3 for a description of the Global Reset, and how it differs from other resets), the 68030 reads the states of two jumper blocks on the 4200/01A's VME board (JB2 and JB3 - see section 2.2.2). The result of this read operation directs the 68030 to one of four tables in Flash Memory. The default contents of these memory tables are described in Table 3-3, below. Table 3-4, also below, gives the access regions for each slave resource, relative to the base addresses and address spaces selected. The uppermost SRAM address will obviously depend upon the amount of memory your 4200/01A contains. Table 3-4 assumes 4 Mbytes are installed. For units with 2 Mbytes of SRAM, the highest SRAM address will be SRAM_base + 0x001FFF FFFF, and for units with 1 Mbyte, SRAM will top out at 0x000F FFFF. The contents of the four base address tables can be modified. Some methods of doing so were discussed in Section 2.4 of this manual.

Table 3-3
Pentek Model 4200/01
Default VME Slave Base Address Table
JB2 and JB3, VME Board

Jumper Positions		SRAM & MIX Address Space	SRAM Base Address	MIX Base Address	ICF Regs (A16)
JB3	JB2				
2 - 3	2 - 3	A32	0x0000 0000	0x4000 0000	0x0000
2 - 3	1 - 2	A32	0x8000 0000	0xC000 0000	0xA000
*1 - 2	2 - 3	A24	0x00 0000	0x40 0000	0x8000
1 - 2	1 - 2	A24	0x80 0000	0xC0 0000	0xC000

* - Factory Default Settings

Table 3-4
Pentek Model 4200/4201
VME Slave Access Memory Map

Address Space	SRAM Region	MIX Module 0	MIX Module 1	MIX Module 2	ICF Region
A32	SRAM_Base + 0x0000 0000 - SRAM_Base + 0x003F FFFF	MIX_Base + 0x0000 0000 - MIX_Base + 0x0FFF FFFF	MIX_Base + 0x1000 0000 - MIX_Base + 0x1FFF FFFF	MIX_Base + 0x2000 0000 - MIX_Base + 0x2FFF FFFF	A16_Base + 0x0000 - A16_Base + 0x00FF
A24	SRAM_Base + 0x00 0000 - SRAM_Base + 0x3F FFFF	MIX_Base + 0x00 0000 - MIX_Base + 0x0F FFFF	MIX_Base + 0x10 0000 - MIX_Base + 0x1F FFFF	MIX_Base + 0x20 0000 - MIX_Base + 0x2F FFFF	A16_Base + 0x0000 - A16_Base + 0x00FF

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2 Model 4200/01A Memory Maps (continued)

3.2.1 VME Slave Memory Map (continued)

From Table 3-3, on the previous page, it is obvious that Jumper Block JB3 assigns the Dual Port SRAM and the MIX bus to either A24 space (with the jumper between pins 1 & 2) or A32 space (with the jumper between pins 2 & 3). It may not be immediately clear what JB2's function is, however. JB2 selects a set of table indices that point to specific locations in the User/Configuration Flash EEPROM to be used as base addresses for the appropriate resources. Table 3-5, below, tells which Flash Memory location is used to store the base address for which resource as a function of where the Jumper Blocks are set.

The Flash Memory locations listed Table 3-5 are the ones that are overwritten when the Monitor program is used to re-program the Slave Base Addresses, as described in Section 2.4.4. To re-program these addresses without using the monitor, three other methods are provided. One of these methods involves erasing and re-programming the 4200/01A's User/Configuration Flash EEPROM, which is the topic covered in Section 3.3 of this manual. Another uses the Host interface program described in Section 2.4.14.1. The other involves writing the desired A32 or A24 base addresses to prescribed locations in A16 space, and is described in Section 2.4.14.2.

Table3-5
Pentek Model 4200/01
EEPROM Table Locations of VME Slave Base Address
JB2 and JB3, VME Board

Jumper Positions		Address Space	SRAM_Base in EEPROM	MIX_Base in EEPROM	ICF (A16_Base) in EEPROM
JB3	JB2				
2 - 3	2 - 3	A32 A16	0xFF07 F000 -	0xFF07 F004 -	- 0xFF07 F200
2 - 3	1 - 2	A32 A16	0xFF07 F008 -	0xFF07 F00C -	- 0xFF07 F204
*1 - 2	2 - 3	A24 A16	0xFF07 F100 -	0xFF07 F104 -	- 0xFF07 F208
1 - 2	1 - 2	A24 A16	0xFF07 F108 -	0xFF07 F10C -	- 0xFF07 F20C

* - Factory Default Jumper Settings

3.2.2 68030 Memory Map

Programs running on the 68030 have access to the VMEbus, the MIX bus, and the VSBus (on a Model 4200A). The 68030's default map of the regions of memory in which these resources are available is shown at the top of the next page, in Table 3-6. This is also the memory map that will be used by an Upper Mix Bus Master when accessing the 4200/01A's resources.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2 Model 4200/01A Memory Maps (continued)

3.2.2 68030 Memory Map (continued)

Table 3-6
Pentek Model 4200/01
68030 Memory Map

Region	Resource
0x0000 0000 - 0x003F FFFF	4 MB SRAM (Dual-Ported w/VME)
0x0040 0000 - 0xBEFF FFFF	A32/D32 VMEbus Master Access
0xBF00 0000 - 0xBFFF FFFF	D32 VSBbus Master Access (4200 Only)
0xC000 0000 - 0xEFFF FFFF	Mix Module Access
0xF200 0000	MIX Flag Register Access
0xF400 0000 - 0xF4FF FFFF	A24/D32 VMEbus Master Access
0xF500 0000 - 0xF5FF FFFF	A24/D16 VMEbus Master Access
0xFF00 0000 - 0xFF03 FFFF	256 kB Flash EEPROM #1 (Monitor)
0xFF04 0000 - 0xFF07 FFFF	256 kB Flash EEPROM #2 (Config.)
0xFFF2 0000	MIX Control Register Access
0xFFFC 0000 - 0xFFFC 00FF	VIC Register Access
0xFFFD 0000 - 0xFFFD 2900	VAC Register Access
0xFFFE 0000 - 0xFFFE FFFF	A16/D32 VMEbus Master Access
0xFFFF 0000 - 0xFFFF FFFF	A16/D16 VMEbus Master Access

At the time of this writing, the only modification that can be made to this memory map is changing the size the 4200A's VSB access region, by decreasing or increasing the A32/D32 VME Master access region. One method available for accomplishing this is via the Monitor program, as described in Section 2.4.5. Another is to manually re-program the User/Configuration Flash EEPROM, as described in Section 3.3, later in this chapter. A third method, available over the VMEbus, is described in Section 2.4.14.2.

3.2.3 The MIX Control Register

The bits in the MIX Control Register, accessible by any local bus master (i. e., an Upper or Lower MIX Bus Master, the 68030 or the VIC) at address 0xFFF2 0000, are identified in Table 3-7, below. Functional descriptions of the register's bits can be found starting at the top of the following page.

Table 3-7
Pentek Model 4200/01 - MIX Control Register - Address = 0xFFF2 0000 - Read & Write Access

Bit	**D23**	**D22**	**D21**	**D20**	**D19**	**D18**	**D17**	**D16**
Name	N/U	N/U	N/U	MIXSplit	SDOut	MXID2*	MXID1*	MXID0*
Bit	**D15**	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**
Name	N/U	N/U	N/U	N/U	N/U	EECS2*	EECS1*	EECS0*
Bit	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Name	TurboMIX	MXMIO*	MXDC*	VMESplit	SDIn	SDClk	Clear Bus	MXRST*

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2 Model 4200/01A Memory Maps (continued)

3.2.3 The MIX Control Register (continued)

The eight upper bits in this 24-bit register (D23 - D16) are Status bits and are Read-Only. The three most significant bits (D23 - D21) are unused. The five remaining status bits are described below.

Bit	Name	Function
D20	MIXSplit	A '1' in this bit indicates that the MIX bus is operating in the "split" configuration. See Section 3.9.3 for further details.
D19	SDOut	This signal line is used by MIX baseboards to read serial data from the User/Configuration EEPROM found on some MIX modules to the baseboard. After writing to a module's EEPROM, this line may be monitored to check the ready status of the EEPROM. The EEPROM is ready when this line returns to the '1' state.

Bit	Name	Function
D18 D17 D16	MXID2* MXID1* MXID0*	These three bits are the MIX module ID bits for modules 2, 1 and 0, respectively. They are read during a MIX bus reset to determine the presence of a module in a given stack position. Installed modules pull these lines to the '0' state when MXRST* is asserted.

The lower 16 bits of the MIX Control Register (D15 - D0) are Bus Control signals and are Read and Write accessible. D15 - D11 are unused. The remaining eleven bits are defined on the following page.

Bit	Name	Function
D10 D9 D8	EECS2* EECS1* EECS0*	These three bits are the EEPROM chip select bits for modules 2, 1 and 0, respectively. They are used by the MIX baseboard to serially transfer configuration data to MIX modules equipped with configuration EEPROMs.
D7	TurboMIX	Writing a '1' to this bit allows the 4200/01A to violate the MIX spec by operating the bus at higher-than-specified speeds.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2 Model 4200/01A Memory Maps (continued)

3.2.3 The MIX Control Register (continued)

Bit	Name	Function
D6	MXMIO*	MIX bus Masters should set this bit to the '0' state for memory transactions with MIX Slave modules, and to the '1' state for I/O transactions. For use with Pentek MIX Modules, set this bit to '0'.
D5	MXDC*	MIX bus Masters should set this bit to the '0' state for memory or I/O transactions with MIX Slave modules, and to the '1' state for IACK and code fetch cycles. For use with Pentek MIX Modules, set this bit to '0'.
D4	VMESplit	Write a '1' to this bit to operate the VMEbus in the "split" configuration. See Section 3.9.3 for further details.
D3	SDIn	This signal line is used by MIX baseboards to write serial data to the configuration EEPROM found on some MIX modules to the baseboard.
D2	SDClk	This is the clock signal used along with the SDIn and SDOut signals to synchronize the serial data transfers with the module EEPROMs.
D1	Clear Bus*	This bit may be set to the '0' state by the 68030 after it has obtained mastership of the local bus, in order to lock out other potential local bus masters. The 68030 will retain local bus mastership until it returns this bit to the '1' state.
D0	MXRST*	The MIX bus, and all the modules on it, will be held in the reset state as long as this bit is in the '0' state.

3.2.4 The MIX Flag Register

This 16-bit register is Read/Write accessible by Upper MIX Bus Masters ONLY, at MIX address 0xF200 0000. Table 3-8, on top of the next page, gives the names of the bits it contains. D15 - D13 and D7 - D4 are unused. The functions of the remaining bits are described below the table.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2 Model 4200/01A Memory Maps (continued)

3.2.4 The MIX Flag Register (continued)

Table 3-8
Pentek Model 4200/01 - MIX Flag Register - Address = 0xF200 0000 - Read & Write Access

Bit	D15	D14	D13	D12	D11	D10	D9	D8
Name	N/U	N/U	N/U	SIZ1	SIZ0	FC2	FC1	FC0
Bit	D7	D6	D5	D4	D3	D2	D1	D0
Name	N/U	N/U	N/U	N/U	MIXSplit	MXINT2EN	MXINT1EN	MXINT0EN

Bit	Name	Function
D12 D11	SIZ1 SIZ0	These two bits are used to indicate the size of the data transaction requested according to the chart below. They are connected to the VIC and the 68030, and can be either inputs or outputs from both those devices, just as they can on the MIX bus.

SIZ0	SIZ1	Transfer
0	0	Longword
0	1	Byte
1	0	Word
1	1	3-byte

D10 D9 D8	FC2 FC1 FC0	These three bits are the Function Code bits, which connect to the 68030, the VIC, and the VAC. These bits are used to determine the target address space for each transaction, according to the chart below.

FC2	FC1	FC0	Transaction
0	0	0	Undefined
0	0	1	User Data Space
0	1	0	User Program Space
0	1	1	Undefined
1	0	0	Undefined
1	0	1	Sup. Data Space
1	1	0	Sup. Program Space
1	1	1	CPU Space

D3	MIXSplit	MIX bus Masters should set this bit to the '1' state in order to split the MIX bus and local bus. See Section 3.9.3 for further details.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2 Model 4200/01A Memory Maps (continued)

3.2.4 The MIX Flag Register (continued)

Bit	Name	Function
D2	MXINT2EN	These three bits are the MIX Interrupt Enable bits, for the modules in stack positions 2, 1 and 0, respectively. The MIX bus Master writes a '1' to the appropriate bit to allow that module to become a MIX bus interrupter.
D1	MXINT1EN	
D0	MXINT0EN	

3.2.5 The MIX Status Register

This register could be regarded as the upper eight bits of the flag register, as it resides at the same address and uses higher-order bits than those contained in the flag register. Access to these bits is Read-Only, however, and so we choose to treat them as a separate entity. Table 3-9, below, summarizes the Status Register's bit structure, and further detail about the relevance of these bits can be found in the paragraphs below the table.

Table 3-9
Pentek Model 4200/01 - MIX Status Register - Address = 0xF200 0000 - Read-Only Access

Bit	D23	D22	D21	D20	D19	D18	D17	D16
Name	LTCHC&S*	LTCHDTA*	DEDLK*	LBERR*	MIXERR*	IPL2*	IPL1*	IPL0*

Bit	Name	Function
D23	LTCHC&S*	These two bits are used in conjunction with the split bus operating mode. See Section 3.9.3 for further details.
D22	LTCHDTA*	
D21	DEDLK*	A '0' in this bit indicates that a deadlock condition has occurred. This is usually the result of an Upper MIX Bus Master (UMBM) attempting to gain mastership of the VMEbus, while an external VMEbus Master is attempting to address the 4200/01A as a Slave.
D20	LBERR*	A '0' in this bit indicates that a local bus error has occurred. These are generally the result of a VMEbus error happening while a UMBM is also a VME Master.
D19	MIXERR*	A '0' in this bit indicates that a MIX bus error has occurred. An example of a MIX bus error would be an attempted unaligned MIX transfer.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.2 Model 4200/01 Memory Maps (continued)

3.2.5 The MIX Status Register (continued)

Bit	Name	Function
D18 D17 D16	IPL2* IPL1* IPL0*	These three bits are the Interrupt Priority level bits, which are driven by the VIC, and also received by the 68030. These bits are the method by which the VIC can generate interrupts to the 68030 and UMBMs.

3.2.6 Access to the VIC and VAC Registers from the 68030

The information from the Cypress Semiconductor VIC/VAC Manual, included here as Appendix A, shows that the registers contained in the VIC068A are all byte-wide (i. e., 8-bit) registers, aligned on odd-word addresses (addresses ending in 3, 7, B or F). However, due to hardware constraints in the design of the Model 4200/01A, it became necessary to treat them as longword (i. e., 32-bit) registers, aligned on longword addresses (addresses ending in 0, 4, 8 or C).

To access any register in the VIC, the 68030 uses the address prefix 0xFFFC in the upper 16 address bits. For the lower 16 address bits, use the nearest longword address boundary LESS THAN the address given for that register in Appendix A. For example, to access the VIC's Block Transfer Definition Register (BTDR), listed in Appendix A at address $AB, the 68030 would use address 0xFFFC 00A8.

Data to be written to any VIC register should be formatted as a 32-bit longword, with the actual data justified in the Least Significant Bits (D7 - D0). Thus, if your 4200/01A is configured as a Slot 1 VMEbus System Controller, and you wanted to reset the VME card cage, you would write the data 0x0000 00F0 to the System Reset Register (SRR) at address 0xFFFC 00E0.

Registers in the VAC068A are also accessible from the 68030. Appendix A lists the addresses of the VAC registers in the form FFFD YZxx, where Y and Z are valid Hex digits, and x is a "don't care" digit position. In the Model 4200/01A, we replace the 'xx' with 00, and write the data as an unsigned short.

The VIC and VAC registers may also be accessed by Upper MIX Bus Masters (UMBMs). Refer to the Operating Manual for the MIX module for information about address translation.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.3 Modifying the Contents of the Flash EEPROMs

In this section, procedures for re-programming the Flash Memories without the use of the Monitor program will be presented. These will include a method for saving changes made to the address tables in the User/Configuration Flash EEPROM, and an explanation of how one can store an application program in the Flash Memory. In all cases, the process consists of four steps, which are: (1) Copy the current contents of the EEPROM to a specific SRAM location; (2) Make the desired changes to the SRAM copy of the code; (3) Erase the Flash EEPROM; (4) Burn the modified copy of the code from the SRAM into the Flash EEPROM.

THE PROCEDURES THAT FOLLOW SHOULD NOT BE UNDERTAKEN BY THE CASUAL USER!! The Flash EEPROMs in the Model 4200/01A contain the device's boot code as well as configuration information and the Monitor program supplied by Pentek. The device can be rendered totally inaccessible by an improper overwrite of the EEPROMs. Proceed at your own risk.

3.3.1 Saving Changes to the Address Tables

Before describing this process, it is worth mentioning that it is considerably easier to do this with the use of the Monitor Program, as described in Section 2.4. This procedure is included in case the requirements of your application preclude the use of the Monitor. Additionally, the EEPROMs must be Write-Enabled before their contents can be changed. See Section 2.2.4 for details.

Begin by copying the entire contents of the User/Configuration Flash EEPROM, (from 68030 address 0xFF04 0000 to 0xFF07 FFFF) into SRAM, beginning at 68030 address 0x0001 0000. Any changes you wish to make to the address configuration should be made to this SRAM copy of the code. The code fragment shown at the top of the next page may be used to copy the EEPROM contents to any SRAM location.

```
#DEFINE Rom (unsigned char *) 0xff040000
            /* Beginning of User/Configuration EEPROM */
CopyRom(unsigned char *Buf)
{
   unsigned long addr;
   unsigned char rom;
   for(addr = 0; addr < CODESIZE; addr++)
      {
      rom = Rom[addr];
      Buf[addr] = rom;
      }
}
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.3 Modifying the Contents of the Flash EEPROMs (continued)

3.3.1 Saving Changes to the Address Tables (continued)

Table 3-5, in Section 3.2.1, gives the EEPROM table locations of the VMEbus Slave resource base addresses. To translate these EEPROM addresses in order to locate the data in the SRAM copy, simply change the 0xFF07 xxxx addresses given in the table to 0x0004 xxxx (this assumes you copied the EEPROM into SRAM beginning at address 0x0001 0000, as suggested at the beginning of this section). Other parameters that can be changed in these tables are the size of the VSB access region, at EEPROM address 0xFF07 F320 (copied to SRAM address 0x0004 F320), the VMEbus arbitration method (0 = Round Robin, 1 = Priority), at EEPROM address 0xFF07 F324 (copied to SRAM address 0x0004 F324), and the VME Bus Request Level (0 - 3, corresponding to BREQ0 - BREQ3) at EEPROM address 0xFF07 F328 (copied to SRAM address 0x0004 F328). The SRAM addresses stated again assume you copied from PROM into SRAM starting at address 0x0001 0000.

After you have made the necessary changes to the SRAM copy of the EEPROM code, the EEPROM can be erased, and the erasure may be verified, using the code fragments beginning below and continuing on the next page.

```
#DEFINE Rom (unsigned char *) 0xff000000
                    /* Offset is 0x0 for Monitor PROM,
                                0x40000 for User PROM */
Erase(unsigned long offset)
{
   Rom[0+offset] = 0xff;   /* reset */
   Rom[0+offset] = 0xff;

   printf("\nErasing Device...");
   Rom[0+offset] = 0x30;
   Rom[0+offset] = 0x30;
   while(!(Rom[0+offset] & 0x80));
   printf("Done\n");
}

VerifyErase(unsigned long offset)
{
   unsigned long addr;
   unsigned char rom;


   printf("\nVerifying Erase...");
   Rom[offset+0] = 0;
   delay(0x10);
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.3 Modifying the Contents of the Flash EEPROMs (continued)

3.3.1 Saving Changes to the Address Tables (continued)

```
   for(addr = 0; addr < ROMSIZE; addr++)
   {
      rom = Rom[offset+addr];
      if(rom != 0xff)
      {
      printf("Erase Failed at %08lx ",&Rom[offset+addr]);
      printf("is %02x should be 0xff\n",rom);
      }
   }
   printf("Done\n");
}
```

Finally, the updated code from SRAM can be programmed into the Flash EEPROM, and the programming can be verified, using the code fragments that follow.

```
#DEFINE RamBuf (unsigned char *)
                /* insert SRAM Address in line above */
BlowRom(unsigned long offset)
{
   unsigned long addr;
   unsigned char rom,ram;
   unsigned int pass;
   printf("\nNow programming rom...");
   Rom[0+offset] = 0xff;      /* reset command */
   Rom[0+offset] = 0xff;

   for(addr = 0; addr < CODESIZE; addr++)
      {
      Rom[0+offset] = 0x50;
      Rom[addr+offset] = ram = RamBuf[addr];
      delay(0x2);
      rom = Rom[addr+offset];

      while(((rom=Rom[addr+offset])&0x80)!=(ram & 0x80))
         {
         if(rom & 0x20)
            {
            printf("Timeout @ addr = %x\n",addr);
            if((Rom[addr+offset]&0x80)!=(ram&0x80))
               {
               printf("\nFailed!!!\n");
               return;
               }
            }
         }
      }

   printf("Done\n");
}
```


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.3 Modifying the Contents of the Flash EEPROMs (continued)

3.3.1 Saving Changes to the Address Tables (continued)

```
Verify(unsigned long offset)
{
   unsigned long addr;
   unsigned char rom, ram;

   printf("\nVerifying ...");
   Rom[0+offset] = 0;      /* read mode */
   delay(0x1);    /* Wait 6 us */

   for(addr = 0; addr < CODESIZE; addr++)
   {
      rom=Rom[addr+offset];
      ram=RamBuf[addr];
      if(rom!=ram)
      {
      printf("Verify failed at %08lx ",addr);
      printf("is %02x should be %02x\n",rom,ram);
      }
   }
   printf("Done\n");
}
```

3.3.2 Writing Application Code into the Flash EEPROMs

The code fragments shown on the previous pages can also be utilized to store your own application code into the 4200/01A's Flash EEPROMs. Your code should be compiled in Motorola Hex format. At the time of this writing, code may be written to the User/Configuration EEPROM only. Care should be taken, however, not to overwrite the address tables in the Configuration EEPROM. In other words, addresses above 0xFF07 F000 (0x0004 F000 in the SRAM copy, again assuming your copy begins at 0x0001 0000) should not be overwritten with application code.

Finally, we will again mention that the Monitor firmware provides a far easier and less risky means of sending user code to the Flash EEPROMs. If possible, we strongly recommend you send your code that way. Refer to Section 2.4.8.2 for details.

3.4 Slave Operation

The sections below describe the operating characteristics of the Model 4200/4201A when it is acting as a Slave device in data transactions over the VMEbus and the MIX bus. No Slave interface to the VSBus is provided on the Model 4200A.

Slave devices are also, generally, interrupt generators. The topic of interrupt generation will be covered later, in Section 3.7.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.4 Slave Operation (continued)

3.4.1 The 4200/01A as a VMEbus Slave

As mentioned in Section 3.2.1, above, there are three resources on the Model 4200/01A that are available for VME Slave Access. These are the ICF Registers, the Dual-Ported SRAM (access to this resource is shared with the 68030), and the MIX Bus. For information about the memory regions occupied by these resources, see Section 3.2.1.

When the VAC detects a VME Address Strobe accompanied by an address within one of these regions (and an appropriate Address Modifier), it asserts one of three selection signals. ICFSEL* is asserted if the access is within the A16 region associated with the Inter-processor Communication Flag Registers. SLSEL0* is asserted for SRAM accesses in A32 or A24 space, and SLSEL1* is asserted for A32 or A24 MIX Bus accesses.

In response to the assertion of any of these signals, the VIC requests control of the local bus. When the local bus Arbiter grants the bus to the VIC, it holds the bus until the completion of the VME transaction, holding off requests from the 68030 or Upper MIX Bus Masters.

VMEbus Slave Block Transfers are implemented on the 4200/01A to the SRAM only. This is because there is no provision made in the MIX bus specification for Slave Block Transfers.

The configuration registers also allow a programmable delay from DSACK* to DTACK*, as well as programmable PAS* and DS* timing. Slave Write-Posting is also supported. When this bus decoupling mode is enabled, the VIC latches data from the VMEbus Master and immediately issues a DTACK*, which keeps the Master from having to wait for local bus availability.

3.4.2 The 4200/01A as a MIX Bus Slave

An Upper Mix Bus Master (UMBM) makes use of the same MIX bus access region as the 68030. The Slave resources available to a UMBM on the 4200/01A are the SRAM, the ICF Registers, the other VIC/VAC registers, the VSBus and the VMEbus.

When a UMBM asserts his MXHLD* signal (the MIX equivalent of a Bus Request), and is granted the local bus by the Arbiter (which asserts MXHLDACK), the module assumes direct mastership of the 4200/01A's local bus. This is in contrast to an external VMEbus Master, who transacts with the local bus via the VIC.

The MIX cycle then proceeds with the assertion of MXCYC* being translated into a local PAS* by means of PALs. After the data is exchanged, the local DSACK* results in the de-assertion of MXCYC*, completing the cycle.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.4 Slave Operation (continued)

3..4.2 The 4200/01A as a MIX Bus Slave (continued)

The MIX specification makes no provision for Slave Block Transfers. However, if a MIX bus Master is capable of holding the MXCYC* line asserted, while toggling MXCMD, it can produce Master Block Transfers.

3.5 Master Operation

The following sections describe the various Bus Master interfaces available on the Models 4200A and 4201A. In addition to its Bus Mastership capabilities on the VME and MIX Busses (and on the VSBus in the 4200A), provisions also exist for a UMBM to assume Mastership of the VMEbus (and the VSBus on the 4200A). Interrupt Handling, also a traditional Bus Master function, is covered in Section 3.8 of this manual.

3.5.1 The 4200/01A as a VMEbus Master

When a local bus resource on the 4200/01A makes a request for a VMEbus Master transaction, the VAC decodes the request and passes it on to the VIC. The VIC then sends a Bus Request to the System Controller (possibly itself), on any of the four levels. When the Bus Grant is received, the VIC performs the transfer and acknowledges the local resource, completing the cycle.

By default, the Model 4200/01A is a level 3 bus requester. One method for changing the requester level, by using the Monitor, was described in Section 2.4.6 (VME Attributes). Changing the requester level from the VMEbus is discussed in Section 2.4.14.3.

The VIC supports Master operations in A16, A24 or A32 space, and can operate in any of the following bus release modes; Release On Request (ROR), Release When Done (RWD), Release on Clear (ROC), Release under RMC* control, and Bus Capture & Hold (BCAP). It is capable of encoding appropriate Address Modifiers from the states of the Function Code and ASIZ* signals, and from the Block Transfer status.

Master Write-Posting is also supported by the VIC. When this bus-decoupling function is enabled, data from the slave is latched by the VIC, and DSACK* is generated immediately, freeing the local bus from VME arbitration delays.

Resources available on the VIC also allow for two types of indivisible data cycles. It can request the VMEbus on the assertion of the RMC* signal for Read-Modify-Write (RMW) operations, independent of the MWB* (Module Wants Bus) signal. This inhibits Slave accesses during RMW cycles. For Indivisible Multiple-Address Cycles (IMACs), the VIC can extend the VMEbus AS* signal.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.5 Master Operation (continued)

3.5.2 The 4200/01A as a MIX Bus Master

The 4200/01A is a MIX baseboard, and as such must provide the bus arbitration functions required in any multi-master bus system. Since the MIX bus is treated as an extension of the 4200/01A's local bus, the local bus arbiter provides this function. In other words, any device that obtains mastership of the 4200/01A's local bus is also master of the MIX bus, and vice versa.

When the 4200/01A functions as master of the MIX bus, (i. e., a Lower MIX Bus Master (LMBM)), either the 68030 or the VIC can be the actual master of the data transactions. Generation of the required Bus Master signals (MXCYC*, MXCMD*, response to MXWAIT*, etc.) is provided by the Baseboard MIX Access PAL.

The unique combination of bus interfaces on the Model 4200/01A also gives an Upper MIX Bus Master (UMBM) the ability to master the VMEbus (or the VSBus on the 4200A). The subsections below describe the mechanics of these master accesses.

3.5.2.1 The 4200/01A's MIX Interface as a VMEbus Master

As stated above, a UMBM that has gained mastership of the MIX bus also has master control of the 4200/01A's local bus, via the Baseboard MIX Access PAL. Any local bus master can become a VMEbus Master, via the VIC. Thus, any UMBM on the 4200/01A baseboard can obtain VMEbus Master status.

For more detail about the mechanics behind a UMBM becoming a VMEbus Master, refer to the Operating Manual for the MIX module in question.

3.5.2.2 The 4200A's MIX Interface as a VSBus Master

Once again, because a UMBM is also master of the 4200A's local bus, it has master access to any device connected to the local bus. This includes the PLX VSB1400A/B chip set, through which VSBus Mastership may be obtained. See the section below for details about the 4200A's VSBus Master operation.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.5 Master Operation (continued)

3.5.3 The 4200A as a VSBus Master

VSB Master access is initiated when a local bus master requests the VSBus, and the request is granted by the VSB Arbiter (which may be the 4200A). When the bus is available, the PLX VSB1400A/B asserts an Address Enable signal (ADDEN*), as well as the BUSY* and HAVEBUS* signals. All three of these signals remain asserted until the completion of the data cycle. The local master then places the address on the local address bus. The PLX chip set reads the address, moves it on to the VSB's common Address/Data bus and asserts the PAS*. ADDEN* is held asserted until the Slave returns an Address Acknowledge signal.

After ADDEN* is de-asserted, the data cycle begins with the assertion of a Data Buffer Enable signal (DBEN*), while the local master puts the data on the local data bus. The PLX chips read the data, move it onto the VSBus, and assert DS*. The data strobe remains asserted until a Data Acknowledge signal is received from the Slave.

3.5.4 DMA and the Model 4200/01A

DMA transactions involving resources on the Model 4200/4201A can be mastered by the VIC/VAC chip set, the 68030, Upper MIX Bus Masters or other VMEbus Masters, depending upon the resources involved. Of these choices, the hardware DMA implemented by the VIC/VAC set is the usually faster than the 68030, which utilizes firmware routines to accomplish DMA-like transfers.

A DMA transaction mastered by the VIC/VAC chip set always has local memory space as either the source or the destination of the data to be transferred. VMEbus DMA is a dual address operation in which the source and destination are required to be on opposite sides of the VME Interface. Off-board VME Masters can also master these transactions. The DMA rate obtainable with an external master is a function of how DMA is carried out by that device.

Upper MIX Bus Masters can master DMA transactions from the MIX bus to VME, from MIX to VSB, from MIX to SRAM or (because a UMBM is also a local bus master) from SRAM to VSB.

The 68030 can master DMA from MIX to SRAM, from VSB to MIX, or from SRAM to VSB. Note that the Model 4200A's VSB interface permits single-cycle DMA access only, and not DMA block transfers.

See the Specifications in Section 1.4 for information about the DMA rates obtainable in all of these configurations. See the programming examples on the enclosed diskette for examples of performing DMA with the Model 4200/01A. See Appendix B of this manual for more information about the diskette.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.6 System Controller Operation

The Model 4200/01A can be configured as a VMEbus Slot 1 System Controller. See Section 2.2.1 for details about the jumper settings that enable this feature. A 4200A can also be configured as a VSB System Arbiter. Section 2.2.3 tells how to set jumpers to enable this feature. The sections below describe the operating characteristics of the 4200A and 4201A as a System Controller/Arbiter.

3.6.1 The 4200/01A as a VMEbus System Controller

VME System Control features on the Model 4200/01A are provided by the VIC. As a controller, the VIC can drive the SYSCLK line on the bus, and can also drive the daisy chained lines, i. e., IACK* and the four Bus Grant I/O levels.

A VME System Controller must also provide bus arbitration functions. As a Bus Arbiter, the VIC can be configured to support Prioritized or Round-Robin arbitration schemes. Round-Robin is the default arbitration method. A programmable timer is available as well, to support Fair-Access timeout arbitration.

Section 2.4.6 of this manual (VME Attributes) discusses setting the VMEbus arbitration mode from the Monitor program, and Section 2.4.14.4 discusses how this may be accomplished from the VMEbus.

3.6.2 The 4200A as a VSBus System Arbiter

The PLX VSB1400A/B chip set provides single-level arbitration only on the VSBus. If a bus request is received from one master while another is in control of the bus, the PLX chips will hold the request and assert BUSY* until the conclusion of the transaction in progress.

3.7 Generating Interrupts with the 4200/01A

The Model 4200/01A is capable of generating interrupts on both the VMEbus and the MIX bus. Additionally, facilities are provided by which MIX modules may generate interrupts on the VMEbus, as well as to the 4200/01A's 68030. The sections below describe the Model 4200/01A's interrupter capabilities.

3.7.1 The 4200/01A as a VMEbus Interrupter

The VIC accepts seven Local Interrupt Request (LIRQ) signals from sources on the 4200/01A. Three of these come from the MIX modules, one from the VSB, and the other three from the VAC's multiplexed PIO interrupt lines (PIO7, 10 and 11). The VAC's PIO interrupts may come from its internal UART (i. e., the Serial Ports) its internal programmable timer, or from the interrupt line of the External Port Header (i. e., the front panel interrupt button).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.7 Generating Interrupts with the 4200/01A (continued)

3.7.1 The 4200/01A as a VMEbus Interrupter (continued)

Programs running on the 68030 may cause the VIC to generate a VMEbus interrupt. Finally, the VIC can receive and respond to an external VMEbus Master's Interrupt Acknowledge (IACK*) signal, and can pass on the daisy chained IACKin/out signal when another device is interrupting.

3.7.1.1 Interrupting the VMEbus from a MIX Module

By default, the 4200/01A maps the interrupt from MIX module 0 to VME IRQ1, using interrupt vector 0xC1. MIX module 1's interrupt is routed to VME IRQ2, utilizing a vector address of 0xC2. Interrupts from MIX module 2 will become VME interrupts on IRQ level 3, vectored to address 0xC3. Table 3-10, below, summarizes the default MIX to VME interrupt mapping.

Table 3-10: Pentek Model 4200/01 Default MIX to VME Interrupt Mapping

MIX Interrupt	VME IRQ Level	Interrupt Vector
MXINT0	IRQ1	0xC1
MXINT1	IRQ2	0xC2
MXINT2	IRQ3	0xC3

These interrupts may be re-mapped by overwriting the contents of a table beginning at VME address SRAM_base+0xC0. The default contents of this table are listed in Table 3-11, at the top of the next page. Note that in the current revision of 4200/01A firmware (Rev. 2.0), these changes CANNOT be saved in the Flash EEPROM.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.7 Generating Interrupts with the 4200/01A (continued)

3.7.1 The 4200/01A as a VMEbus Interrupter (continued)

3.7.1.1 Interrupting the VMEbus from a MIX Module (continued)

Table 3-11: Pentek Model 4200/01 MIX to VME Interrupt Mapping Table in Model 4200/01 SRAM

Address (SRAM_base+)	Default Contents	Comments
0xC0	0x00	Global wait-until-ACK enable. If set, waits in ISR that asserted the VME IRQ until Global ACK bit is set. This has the effect of disabling any other interrupts until the current service is complete.
0xC4	0x00	Global ACK. If wait-until-ACK mode is set (see above) this bit allows ISR to complete when set by external handler.
0xC8	0x00	Global module ID. Identifies which module (1, 2 or 3) is interrupting.
0xCC	0x01	Module 0 IRQ level
0xD0	0xC1	Module 0 IACK vector
0xD4	0x02	Module 1 IRQ level
0xD8	0xC2	Module 1 IACK vector
0xDC	0x03	Module 2 IRQ level
0xE0	0xC3	Module 2 IACK vector

3.7.1.2 Interrupting the VMEbus from a 68030 Program

A program running on the 68030 can generate a VMEbus interrupt by writing to a register in the VIC. See Appendix A for further details about the VIC registers, and see the programming examples on the enclosed diskette for an example of a 68030 program that interrupts VME. See Appendix B for more information about the diskette.

3.7.2 The 4200/01A as a MIX Bus Interrupter

The Model 4200/01A can issue a MIX interrupt to a UMBM in any of the three stack positions. MIX interrupts may be generated in response to error conditions detected in the MIX Master Status Register, to VME interrupts, or by programs running on the 68030.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.7 Generating Interrupts with the 4200/01A (continued)

3.7.2 The 4200/01 as a MIX Bus Interrupter (continued)

3.7.2.1 Interrupting the MIX Bus from the VMEbus

The 4200/01A, when acting as a VMEbus interrupt handler, can generate MIX bus interrupts. In this situation, the VIC receives one of the seven VME Interrupt Request (IRQ) signals, and asserts one or more of its Interrupt Priority Level (IPL) outputs, depending upon the contents of the VIC's IRQ Control Registers (see Appendix A). These three outputs are connected to the MIX Interrupt PAL. From here, the signal can be routed through to the appropriate MIX interrupt line, under the control of the MIX Flag Register.

3.7.2.2 Interrupting MIX Processor Modules from 68030 Programs

Programs running on the 68030 **CANNOT** generate MIX bus interrupts. A 68030 program may, however, write to the mailbox register on a MIX module, which will interrupt the module's processor.

3.7.2.3 Local Conditions That Will Generate MIX Interrupts

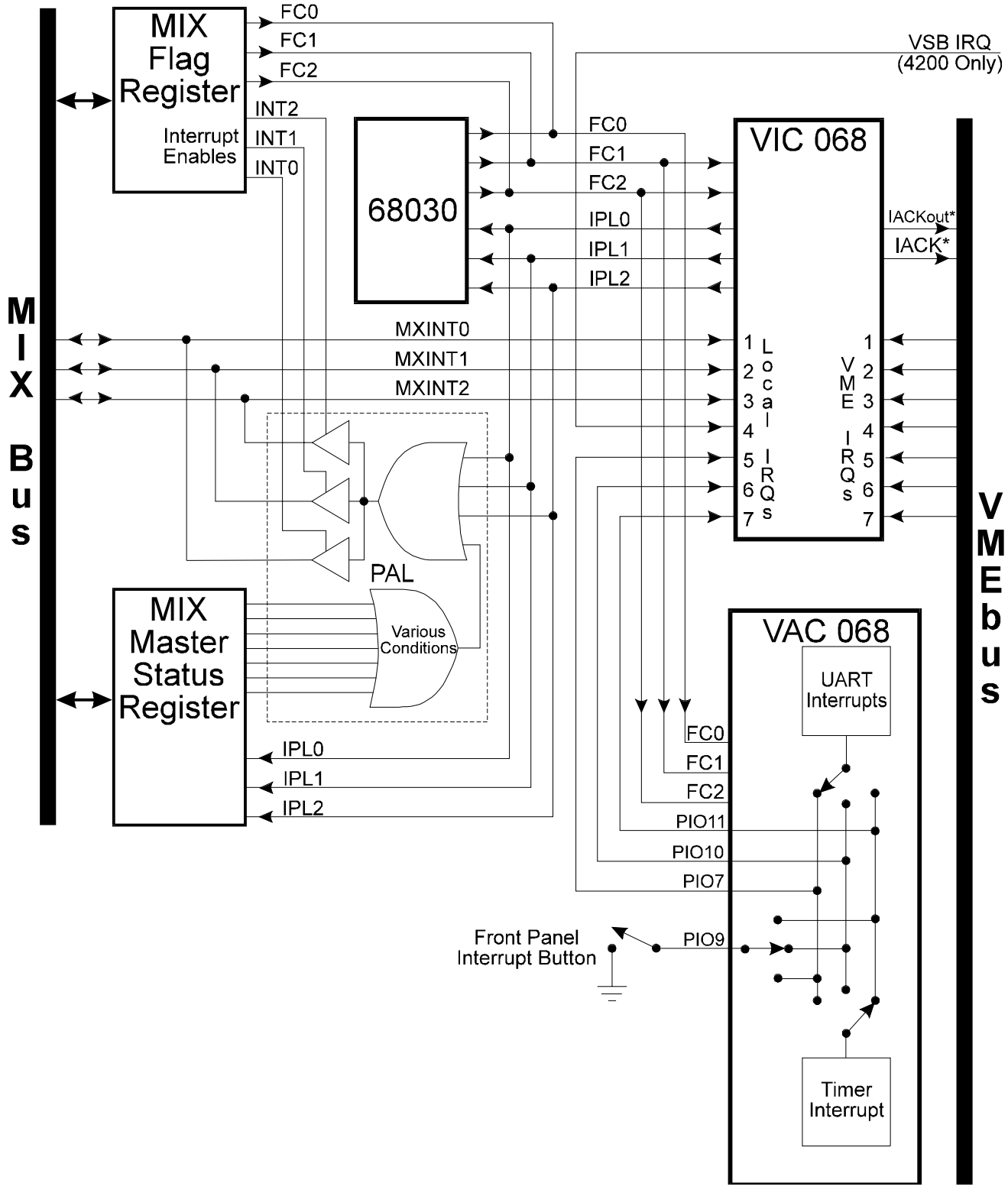
1) Local Bus Error - This is the local representation of a VMEbus error. If a UMBM is mastering the VMEbus, a VMEbus error will result in a Local Bus Error, which will in turn generate a MIX interrupt.

2) Deadlock - A deadlock occurs when a device is attempting to become a VMEbus master while another master is simultaneously addressing him as a slave. This situation can arise when a UMBM asynchronously obtains control of the local bus when an external VMEbus Master has just successfully arbitrated for with the VIC for local bus mastership.

3) Local Mix Error - This is usually the result of an attempt to perform an unaligned transfer on the MIX bus, e. g., a longword transfer on address 0xC000 0001.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Figure 3-3
Pentek Model 4200/4201A
Interrupt Handling
Block Diagram



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.8 Handling Interrupts to the 4200/01A

The Model 4201A can be an interrupt handler on the VMEbus and the MIX bus. The Model 4200A can be a VSBus interrupt handler as well. In some cases, handling an interrupt from one bus involves the generation of an interrupt to the 68030, or to another bus. Figure 3-3, on the previous page, is a block diagram showing 4200/01A's interrupt handling apparatus. The sections that follow describe the operation of the 4200/01A as an interrupt handler.

3.8.1 The 4200/01A as a VMEbus Interrupt Handler

Manipulation of the VIC's interrupt handling configuration registers can cause signals from each of the seven levels of VME Interrupt Requests (IRQs) to activate one or more of the VIC's Interrupt Priority Level (IPL) outputs. The IPL's are connected to both the 68030 and the MIX Interrupt PAL.

3.8.1.1 VMEbus Interrupts to the 68030

The 68030 is interrupted by means of three Interrupt Priority Level (IPL) inputs. The VIC can be configured to drive these inputs in response to any of the seven levels of VMEbus Interrupt Requests (IRQs).

3.8.1.2 VMEbus Interrupts to the MIX Bus

The VIC's Interrupt Priority Level (IPL) outputs are also connected to the MIX Interrupt PAL. Here, the IPL can be routed to any of the three MIX module interrupt lines under the control of the MIX Master Register. One must take care to see that the 68030 does not respond to the IPL he intends for the MIX bus. Under the current release of 4200/01A firmware, the 68030 does not respond to IPL codes 1, 2 or 3, so these may be used to generate MIX interrupts.

3.8.2 The 4200/01A as a MIX Bus Interrupt Handler

The three MIX Module Interrupts (MXINT0, 1 and 2), are connected to the VIC's Local IRQ1, 2 and 3 inputs, respectively. The VIC can respond to these Local Interrupt Requests by passing the interrupt on to the 68030 or the VMEbus, as described in the sections below.

3.8.2.1 MIX Bus Interrupts to the 68030

The VIC can respond to a MIX interrupt, connected to one of its Local Interrupt Request inputs, by asserting its Interrupt Priority Level (IPL) outputs and interrupting the 68030.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.8 Handling Interrupts to the 4200/01A (continued)

3.8.2 The 4200/01A as a MIX Bus Interrupt Handler (continued)

3.8.2.2 MIX Bus Interrupts to the VMEbus

The VIC can respond to a MIX interrupt, connected to its Local Interrupt Request (LIRQ) inputs, by asserting VMEbus IRQs. The 4200/01A's firmware maps MIX interrupts to VMEbus IRQs by a scheme presented in Table 3-9 (see Section 3.7.1.1).

3.8.3 The 4200A as a VSBus Interrupt Handler

The VSBus Interrupt Request (IRQ) signal is connected to the Local IRQ4 input of the VIC. When the VIC receives this signal, it can pass the interrupt on to the 68030 or the MIX bus. Each of these situations is discussed in the sections below.

3.8.3.1 VSBus Interrupts to the 68030

The VIC can respond to the VSBus interrupt, connected to its Local IRQ4 input, by asserting its Interrupt Priority Level (IPL) outputs and interrupting the 68030.

3.8.3.2 VSBus Interrupts to the MIX Bus

The VIC can respond to the VSBus interrupt, connected to its Local IRQ4 input, by asserting its three Interrupt Priority Level (IPL) outputs. The IPLs are connected to the MIX Interrupt PAL. Here, the IPL can be routed to any of the three MIX module interrupt lines under the control of the MIX Flag Register.

3.9 Additional Features of the Model 4200/01A

In addition to the features described above, users of the Model 4200/01A also have access to two resources contained on the VAC, the programmable Timer and the UART (Universal Asynchronous Receiver/Transmitter). The sections beginning on the next page describe the use of these devices.

3.9.1 Using the VAC's Programmable Timer

The VAC's Programmable Timer consists of a 6-bit prescaler whose carry output clocks a 16-bit programmable counter. The prescaler is clocked by the 16MHz CPU clock. Operation of the timer is controlled by two registers in the VAC, the Timer Data Register at address 0xFFFD 2700, and the Timer Control Register at 0xFFFD 2800. When the timer reaches its final value, an interrupt is generated.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.9 Additional Features of the Model 4200/01A (continued)

3.9.1 Using the VAC's Programmable Timer (continued)

The Control Register contains the bits to be loaded into the prescaler, as well as a register at which the prescaler's instantaneous value may be read. Also included are a Run/Load bit, which can either hold the counter in its initial state or allow it to count, and a Once/Continuous bit, which determines whether the counter counts once and stops or counts continuously.

The data register contains the initial value to be loaded into the counter. When this register is read, the counter's instantaneous value is placed on the data bus, not the loaded value.

See Appendix A (VIC/VAC Configuration Registers) for more information about the Timer Data and Control Registers.

3.9.2 Using the VAC's UART

The VAC's UART is a dual-channel, full-duplex device, available to the user at the front panel Serial Port connectors, or at the External Port Header if the front panel is removed. It is serviced via interrupts. The VAC contains ten registers for configuration and control of the UART, five for each channel.

The UART Mode Registers (Channel A's at address 0xFFFD 1D00, Channel B's at 0xFFFD 1F00) allow configuration for looped operation of the transmitter or receiver, allowing or disallowing transmission breaks, enabling or disabling the channel, resetting the channel, setting the BAUD rate and parity generation and checking.

The Transmit Data Registers (0xFFFD 1E00 for Channel A, 0xFFFD 2200 for Channel B) are loaded with the data to be transmitted. A pair of four-byte deep Receiver FIFOs (at 0xFFFD 2000 for Channel A and 0xFFFD 2100 for Channel B) contain reception error bits and the received data.

The Interrupt Mask Registers (Channel A - 0xFFFD 2300, Channel B - 0xFFFD2400) determine the conditions that will cause the UART to generate interrupts. Interrupt Status Registers (at 0xFFFD 2500 for Channel A and 0xFFFD 2600 for Channel B) are read to determine why a particular interrupt was generated.

When a UART interrupt is received, the VAC's main Interrupt Status Register is read to determine which channel generated the interrupt. Next, that channel's Interrupt Status register is read to determine the cause of the interrupt. That interrupt is then masked in the channel's Interrupt Mask register. The interrupt may then be serviced, after which the channel's Interrupt Status Register should be cleared and the Mask register restored to its original state.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

3.9 Additional Features of the Model 4200/01A (continued)

3.9.2 Using the VAC's UART (continued)

See Appendix A (VIC\VAC Configuration Registers) for more information about the UART Registers.

3.9.3 Split Bus Operation

This feature is not yet supported in the current revision of 4200/01A hardware and firmware. Please contact the factory at the number shown on the cover page of this manual for more details about this operating mode and its projected availability date.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

This page is intentionally blank

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Appendix A: Cypress Semiconductor VIC/VAC Data

A.1 VIC068A Interprocessor Communication Facilities

TheVIC068A contains three categories of interprocessor communication facilities (ICFs):

- ❏ Interprocessor Communication Registers (ICRs)
- ❏ Interprocessor Communication Global Switches (ICGSs)
- ❏ Interprocessor Communication Module Switches (ICMSs)

The ICRs are 8-bit registers that may be accessed from either the Local bus or the VMEbus.

The ICGSs and ICMSs are switches that may be set to interrupt the local processor.

These facil;ities are located in seven registers that are visible from both the Local bus and the VMEbus. When accessed via the Local bus, the registers are read/written by normal VIC068A register access methods. When accessed by the VMEbus, the $\overline{\text{ICFSEL}}$ signal is used as a register select signal. The register addresses, when accessed from the Local bus are not the same as when accessed from the VMEbus. The VIC068A contains an internal arbiter to arbitrate between Local and VMEbus accesses to these facilities.

Additional registers used for the ICFs listed below. The contents of these registers are described in detail in the sections numbers following the abbreviated names in the list.

- ❏ ICGS Interrupt Control Register (ICGSICR - A.2.5)
- ❏ ICMS Interrupt Control Register (ICMSICR - A.2.6)
- ❏ ICGS Interrupt Vector Base Register (ICGSIVBR - A.2.8)
- ❏ ICMS Interrupt Vector Base Register (ICMSIVBR - A.2.9)
- ❏ Interprocessor Communication Switch Register (ICSR - A.2.12)

A.1.1 Valid ICF Selection

The ICFSEL signal is used to signal that the VMEbus master desires access to the VIC068A interprocessor communication facilities. This signal is usually driven from VMEbus address decoders. When ICFSEL is asserted, the VIC068A checks A[5:1] to determine what ICF is desired. Self-access to the ICF facilities is not detected by the VIC068A. The VIC068A then verifies the AM codes against an internal table, shown in Table A-1, at the top of the next page.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.1 VIC068A Interprocessor Communication Facilities (continued)

A.1.1 Valid ICF Selection (continued)

Table A-1: VIC068A - Valid ICF Address Modifiers		
ICF	**AM Codes**	**Comments**
ICRs	0x29, 0x2D	A16, User or Supervisory Data
ICGSs	0x2D	A16, Supervisory Data
ICMS	0x29, 0x2D	A16, User or Supervisory Data

Once a valid ICF select has occurred, the VIC068A then processes the request. The ICF VMEbus addresses are shown in Table A-2, below.

Table A-2: VIC068A - ICF VMEbus Address Map	
Function	**Address**
ICR0 Access	A16_base+0x01
ICR1 Access	A16_base+0x03
ICR2 Access	A16_base+0x05
ICR3 Access	A16_base+0x07
ICR4 Access	A16_base+0x09
ICR5 Access	A16_base+0x0B
ICR6 Access	A16_base+0x0D
ICR7 Access	A16_base+0x0F
Clear ICGS0	A16_base+0x10
Set ICGS0	A16_base+0x11
Clear ICGS1	A16_base+0x12
Set ICGS1	A16_base+0x13
Clear ICGS2	A16_base+0x14
Set ICGS2	A16_base+0x15
Clear ICGS3	A16_base+0x16
Set ICGS3	A16_base+0x17
Clear ICMS0	A16_base+0x20
Set ICMS0	A16_base+0x21
Clear ICMS1	A16_base+0x22
Set ICMS1	A16_base+0x23
Clear ICMS2	A16_base+0x24
Set ICMS2	A16_base+0x25
Clear ICMS3	A16_base+0x26
Set ICMS3	A16_base+0x27

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.1 VIC068A Interprocessor Communication Facilities (continued)

A.1.2 Interprocessor Communication Registers

The VIC068A contains eight Interprocessor Communication Registers (ICRs). These registers are accessible from both the Local bus and the VMEbus. ICRs 4 - 0 are considered general purpose read/write registers. ICR5 is the VIC068A revision register. The value read from this register indicates the mask revision of the device. ICR6 contains the $\overline{\text{HALT}}$ and $\overline{\text{RESET}}$ status of the VIC068A. ICR7 provides semaphores for ICRs 5 - 0. These semaphores are set whenever ICRs 5 - 0 are written. In addition, ICR7 also indicates VMEbus mastership and has a mask for $\overline{\text{SYSFAIL}}$. Refer to Sections A.2.13 through A.2.17 for detailed register descriptions.

A.1.3 Interprocessor Communication Global Switches

The ICGSs are software switches that may be set over the VMEbus to interrupt a group of VMEbus modules. When the VIC068A issues the global switches, it is performing a VMEbus byte-wide write to the pre-defined global switch address.

If the global switch interrupts are enabled, in the ICGSICR of the VIC068A slave, a local interrupt is generated on a clear-to-set transition of the selected switch. When acknowledged (FCIACK asserted), the slave VIC068A handles the interrupt by returning the status/ID value from the ICGSVBR. See Chapter 9 of Cypress Semiconductor's **VIC068A/VAC068A User's Guide** for details on VIC068A interrupt generation and handling. Once a switch is set, it must be cleared before it can be set again.

A.1.4 Interprocessor Communication Module Switches

Like the ICGSs, the ICMSs are software switches that may be set over the VMEbus to interrupt the local processor. The module switches, however, are meant to be issued to a specific module. As in the Global switches, the VIC068A issuing the module switches performs a byte-wide write to the pre-defined switch address.

Because the module switches are meant for a specific module, the VIC068A as VMEbus slave (the module whose switch was just set) must assert the $\overline{\text{DTACK}}$. The VIC068A issuing the ICMS need not have its $\overline{\text{ICFSEL}}$ signal asserted.

The interrupt and addressing mechanisms are the same for the ICMSs as they were for the ICGSs. The ICSR may be used to provide monitoring of the module switches. Unlike the global switches, this register may be written by local resources to interrupt the CPU.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions

This chapter describes the VIC068A internal configuration registers. These registers en-able and disable various features of the VIC068A. (Refer to the specific sections of the **VIC068A User's Guide** for details on specific features.)

Table A-3, on the next page, provides information on the various reset states of the VIC068A registers. The following notes should be observed regarding this table:

- An asterisk (*) indicates a bit that is not affected by the particular reset.
- An "X" indicates a bit that is affected by the state of a particular VIC068A pin.
- ICR5 is the VIC068A Version register. Its contents will vary depending on the revision of the device being used.

Unless otherwise specified, the reset status is given in this chapter by the values located in the parentheses below each bit field. The (X/X/X) format indicates the Global/Internal/System reset state of each bit or bit field.

For compatibility with the VIC64, it is recommended that all reserved register bits be written with a '0'.

A.2.1 VMEbus Interrupter Interrupt Control Register

Name: VIICR

Address: 0xFFFC 0003

Description: Provides enabling and IPL level encoding for the local interrupt issued when a VMEbus interrupt is acknowledged.

Bits 2 - 0: (0/*/*) IPL value: Value is inverted and driven onto the IPL lines when an interrupt is acknowledged.

Bits 6 - 3: (1/1/1) Undefined/Reserved. Bits will read as 1's.

Bit 7: (1/1/1) VMEbus interrupt mask: When clear, the VIC068A signals a local interrupt at the acknowledgement of a previously issued VMEbus interrupt. When set, the VIC068A will not issue a local interrupt.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

Table A-3: VIC068A - Register Values After Reset Operations

Address (hex)	Name	Description	Global Reset	Internal Reset	System Reset
0xFFFC 0003	VIICR	VMEbus Interrupter Interrupt Control Register	1111 1000	1111 1***	1111 1***
0xFFFC 0007 - 1F	VICR1 - 7	VMEbus Interrupt Control Registers 1 - 7	1111 1000	1111 1***	1111 1***
0xFFFC 0023	DMASICR	DMA Status Interrupt Control Register	1111 1000	1111 1***	1111 1***
0xFFFC 0027 - 3F	LICR1 - 7	Local Interrupt Control Registers 1 - 7	1000 X000	1*** X***	1*** X***
0xFFFC 0043	ICGSICR	ICGS Interrupt Control Register	1111 1000	1111 1***	1111 1***
0xFFFC 0047	ICMSICR	ICMS Interrupt Control Register	1111 1000	1111 1***	1111 1***
0xFFFC 004B	EGICR	Error Group Interrupt Control Register	1111 X000	1111 X***	1111 X***
0xFFFC 004F	ICGSIVBR	ICGS Interrupt Vector Base Register	0000 11XX	0000 11XX	0000 11XX
0xFFFC 0053	ICMSIVBR	ICMS Interrupt Vector Base Register	0000 11XX	0000 11XX	0000 11XX
0xFFFC 0057	LIVBR	Local Interrupt Vector Base Register	0000 1XXX	0000 1XXX	0000 1XXX
0xFFFC 005B	EGIVBR	Error Group Interrupt Vector Base Register	00001XXX	00001XXX	00001XXX
0xFFFC 005F	ICSR	Interprocessor Communications Switch Register	0000 0000	**** 0000	0000 0000
0xFFFC 0063 - 73	ICR0 - 4	Interprocessor Communications Registers 0 - 4	0000 0000	0000 0000	0000 0000
0xFFFC 0077	ICR5	Interprocessor Communications Register 5	Version	Version	Version
0xFFFC 007B	ICR6	Interprocessor Communications Register 6	X111 11XX	X111 1111	X111 1110
0xFFFC 007F	ICR7	Interprocessor Communications Register 7	00X0 0000	*0XX ****	00X0 0000
0xFFFC 0083	VIRSR	VMEbus Interrupt Request Status Register	0000 0000	**** ***0	0000 0000
0xFFFC 0087 - 9F	VIVBR1 - 7	VMEbus Interrupt Vector Base Registers 1 - 7	0000 1111	**** ****	0000 1111
0xFFFC 00A3	TTR	Transfer Timeout Register	0110 1000	0110 1000	0110 1000
0xFFFC 00A7	LBTR	Local Bus Timing Register	0000 0000	**** ****	**** ****
0xFFFC 00AB	BTDR	Block Transfer Definition Register	1111 0000	1111 0000	1111 0000
0xFFFC 00AF	ICR	Interface Configuration Register	0000 000X	0000 000X	0000 000X
0xFFFC 00B3	ARCR	Arbiter/Requester Configuration Register	0110 0000	011* 0000	011* 0000
0xFFFC 00B7	AMSR	Address Modifier Source Register	0000 0000	0000 0000	0000 0000
0xFFFC 00BB	BESR	Bus Error Status Register	X000 0000	X000 0000	X000 0000
0xFFFC 00BF	DMASR	DMA Status Register	0110 0000	0110 0000	0110 0000
0xFFFC 00C3	SS0CR0	Slave Select 0 Control Register 0	0000 0000	00** ****	00** ****
0xFFFC 00C7	SS0CR1	Slave Select 0 Control Register 1	0000 0000	**** ****	**** ****
0xFFFC 00CB	SS1CR0	Slave Select 1 Control Register 0	0000 0000	00** ****	00** ****
0xFFFC 00CF	SS1CR1	Slave Select 1 Control Register 1	0000 0000	**** ****	**** ****
0xFFFC 00D3	RCR	Release Control Register	0000 0000	0000 0000	0000 0000
0xFFFC 00D7	BTCR	Block Transfer Control Register	0000 0000	0000 0000	0000 0000
0xFFFC 00DB	BTLR0	Block Transfer Length Register 0	0000 0000	0000 0000	0000 0000
0xFFFC 00DF	BTLR1	Block Transfer Length Register 1	0000 0000	0000 0000	0000 0000
0xFFFC 00E3	SRR	System Reset Register	1111 1111	1111 1111	1111 1111
0xFFFC 00EB - FF	—	Reserved Locations	1111 1111	1111 1111	1111 1111

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.2 VMEbus Interrupt Control Registers 1-7

Name: VICR1-7

Addresses	Interrupt
0xFFFC 0007	1
0xFFFC 000B	2
0xFFFC 000F	3
0xFFFC 0013	4
0xFFFC 0017	5
0xFFFC 001B	6
0xFFFC 001F	7

Description: Provides enabling of the VIC068A as VMEbus interrupt handler for any or all of the VMEbus interrupts. Seven registers exist to provide unique masking and IPL values for the seven VMEbus interrupts.

Bits 2 - 0: (0/*/*) IPL value: Value is inverted and driven onto the IPL signals when a VMEbus interrupt is acknowledged.

Bits 6 - 3: (1/1/1) Undefined/Reserved. Bits will read as 1's.

Bit 7: (1/1/1) VMEbus interrupt mask: When clear, the VIC068A acts as a VMEbus interrupt handler by signaling a local interrupt at the specified IPL level. When set, the VIC068A does not handle the VMEbus interrupt and no local interrupt is issued.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.3 DMA Status Interrupt Control Register

Name: DMASICR

Address: 0xFFFC 0023

Description: Provides enabling and IPL-level encoding for the DMA-complete interrupt issued by the VIC068A when any VIC068A local DMA operation completes (successfully or unsuccessfully).

Bits 2 - 0: (0/*/*) IPL value: Value is inverted and driven onto the IPL lines when interrupt is acknowledged.

Bits 6 - 3: (1/1/1) Undefined/Reserved. Bits will read as 1's.

Bit 7: (1/1/1) DMA status interrupt mask: When clear, the VIC068A signals a local interrupt at the completion of any VIC068A local DMA operation. When set, the VIC068A will not issue a local interrupt.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.4 Local Interrupt Control Registers 1-7

Name: LICR1 - 7

Address	LICR
0xFFFC 0027	1
0xFFFC 002B	2
0xFFFC 002F	3
0xFFFC 0033	4
0xFFFC 0037	5
0xFFFC 003B	6
0xFFFC 003F	7

Description: Provides enabling, IPL level, and control of local interrupts 1 - 7 ($\overline{\text{LIRQ1-7}}$).

Bits 2 - 0: (0/*/*) IPL value: Value is inverted and driven onto the IPL lines when a local interrupt is presented on the $\overline{\text{LIRQ1-7}}$ signals and bit 7 of this register is clear (enabled).

Bit 3: (X/X/X) $\overline{\text{LIRQ1-7}}$ voltage state: A cleared bit indicates an $\overline{\text{LIRQ1-7}}$ signal is asserted at the VIC068A.

Bit 4: (0/*/*) Autovector enable: When set, the VIC068A will supply the interrupt status/ID vector for the local interrupt acknowledge cycle. When cleared, the VIC068A will assert the $\overline{\text{LIACK0}}$ signal to indicate a 680x0 "autovector" condition or that the interrupting source should provide the Status/ID vector to the processor.

Bit 5: (0/*/*) Edge/level enable: When cleared, the VIC068A responds to the $\overline{\text{LIRQ1-7}}$ as a level-sensitive interrupt. When set, the VIC068A responds to $\overline{\text{LIRQ1-7}}$ as an edge-sensitive interrupt.

Bit 6: (0/*/*) Polarity set: When set, the VIC068A responds to interrupts as active High if bit 5 is set (level sensitive) or on a rising edge if bit 5 is cleared (edge sensitive). When cleared, the VIC068A responds to active Low or falling edges.

Bit 7: (1/1/1) Local interrupt mask: When clear, the VIC068A is enabled to handle the corresponding local interrupt asserted on the $\overline{\text{LIRQ1-7}}$ signals.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.5 ICGS Interrupt Control Register

Name: ICGSICR

Address: 0xFFFC 0043

Description: Provides enabling and IPL encoding for the four global switch interrupts.

Bits 2 - 0: (0/*/*) IPL Value: Value is inverted and driven onto the IPL signals when a global switch is acknowledged.

Bit 3: (1/1/1) Undefined/Reserved. Bit will read as a 1.

Bit 4: (1/1/1) ICGS0 mask: When clear, the VIC068A will issue and handle a local interrupt when global switch 0 is set.

Bit 5: (1/1/1) ICGS1 mask: When clear, the VIC068A will issue and handle a local interrupt when global switch 1 is set.

Bit 6: (1/1/1) ICGS2 mask: When clear, the VIC068A will issue and handle a local interrupt when global switch 2 is set.

Bit 7: (1/1/1) ICGS3 mask: When clear, the VIC068A will issue and handle a local interrupt when global switch 3 is set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.6 ICMS Interrupt Control Register

Name: ICMSICR

Address: 0xFFFC 0047

Description: Provides enabling and IPL encoding for the four module switch interrupts.

Bits 2 - 0: (0/*/*) IPL Value: Value is inverted and driven onto the IPL signals when a module switch is acknowledged.

Bit 3: (1/1/1) Undefined/Reserved. Bit will read as a 1.

Bit 4: (1/1/1) ICMS0 mask: When clear, the VIC068A will issue and handle a local interrupt when module switch 0 is set.

Bit 5: (1/1/1) ICMS1 mask: When clear, the VIC068A will issue and handle a local interrupt when module switch 1 is set.

Bit 6: (1/1/1) ICMS2 mask: When clear, the VIC068A will issue and handle a local interrupt when module switch 2 is set.

Bit 7: (1/1/1) ICMS3 mask: When clear, the VIC068A will issue and handle a local interrupt when module switch 3 is set.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.7 Error-Group Interrupt Control Register

Name: EGICR

Address: 0xFFFC 004B

Description: Provides enabling and IPL encoding for the error group interrupts.

Bits 2 - 0: (0/*/*) IPL Value: Value is inverted and driven onto the IPL signals when an error group interrupt is acknowledged.

Bit 3: (X/X/X) $\overline{\text{SYSFAIL}}$ asserted: This bit is set whenever $\overline{\text{SYSFAIL}}$ is detected asserted.

Bit 4: (1/1/1) $\overline{\text{SYSFAIL}}$ interrupt mask: When clear, the VIC068A generates a local interrupt when $\overline{\text{SYSFAIL}}$ is asserted.

Bit 5: (1/1/1) Arbitration timeout interrupt mask: When clear, the VIC068A generates a local interrupt when an arbitration timeout has occurred.

Bit 6: (1/1/1) Write post fail interrupt mask: When clear, the VIC068A generates a local interrupt when a write post operation has failed due to a bus error. For master write posts, an assertion of $\overline{\text{BERR}}$ will trigger an interrupt. For slave write posts, an assertion of $\overline{\text{LBERR}}$ will trigger an interrupt.

Bit 7: (1/1/1) AC Fail interrupt mask: When clear, the VIC068A generates a local interrupt when $\overline{\text{ACFAIL}}$ is detected as asserted.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.8 ICGS Interrupt Vector Base Register

Name: ICGSIVBR

Address: 0xFFFC 004F

Description: Provides the status/ID vector for the global switch interrupts. This register must be written after any VIC068A reset to enable identification encoding for bits 1 - 0.

Bits 1 - 0: (X/X/X) Global switch number (read-only): This value indicates which global switch is pending during a global switch interrupt acknowledge cycle. These bits are used with bits 7 - 2 to provide a unique status/ ID vector for each global switch. The numeric value of this field indicates the switch number. These bits are valid only during the interrupt acknowledge cycle.

Bits 7 - 2: (1/1/1 [3-2]) (0/0/0 [7-4]) Status/ID: These bits are user-definable and are used with bits 1 - 0 to provide a unique global switch interrupt status/ID vector.

A.2.9 ICMS Interrupt Vector Base Register

Name: ICMSIVBR

Address: 0xFFFC 0053

Description: Provides the status/ID vector for the module switch interrupts. This register must be written after any VIC068A reset to enable identification encoding for bits 1 - 0.

Bits 1 - 0: (X/X/X) Module switch number (read-only): This value indicates which module switch is pending during a module switch interrupt acknowledge cycle. These bits are used with bits 7 -2 to provide a unique status/ID vector for each module switch. The numeric value of this field indicates the switch number. These bits are valid only during the interrupt acknowledge cycle.

Bits 7 - 2: (1/1/1 [3-2]) (0/0/0 [7-4]) Status/ID: These bits are user-definable and are used with bits 1 - 0 to provide a unique global switch interrupt status/ID vector.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.10 Local Interrupt Vector Base Register

Name: LIVBR

Address: 0xFFFC 0057

Description: Provides the status/ID vector for the local interrupts. This register must be written after any VIC068A reset to enable identification encoding for bits 2 - 0.

Bits 2 - 0: (X/X/X) Local Interrupt number (read-only): This value indicates which local interrupt is pending during a local interrupt acknowledge cycle. These bits are used with bits 7 - 3 to provide a unique status/ ID vector for each local interrupt. The numeric value of this field indicates the local interrupt number. These bits are valid only during the interrupt acknowledge cycle.

Bits 7 - 3: (1/1/1 [3]) (0/0/0 [7-4]) Status/ID: These bits are user-definable and are used with bits 2 - 0 to provide a unique local interrupt status/ID vector.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.11 Error Group Interrupt Vector Base Register

Name: EGIVBR

Address: 0xFFFC 005B

Description: Provides the status/ID vector for the error group interrupts. This register must be written after any VIC068A reset to enable identification encoding for bits 2 - 0.

Bits 2 - 0: (X/X/X) Error/Status Group Interrupt number (read-only): This value indicates which group interrupt is pending during the interrupt acknowledge cycle. These bits are used with bits 7 - 3 to provide a unique status/ID vector for each error group interrupt. These bits are valid only during the interrupt acknowledge cycle.

Bit 2 1 0	Error/Status Interrupt
0 0 0	$\overline{\text{ACFAIL}}$ asserted
0 0 1	Write post failed
0 1 0	Arbitration timeout
0 1 1	$\overline{\text{SYSFAIL}}$ asserted
1 0 0	VMEbus Interrupter interrupt acknowledge
1 0 1	DMA complete
1 1 0	Not used
1 1 1	Not used

Bits 7 - 3: (1/1/1 [3]) (0/0/0 [7-4]) Status/ID: These bits are user-definable and are used with bits 2 - 0 to provide a unique local interrupt status/ID vector.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.12 Interprocessor Communications Switch Register

Name: ICSR

Address: 0xFFFC 005F

Description: Provides setting, clearing, and monitoring of the interprocessor switch interrupts via the local bus. If the switch interrupts are enabled, setting these bits (more precisely, a clear-to-set transition) causes a local interrupt to occur in the same way as if the switch was set over the VMEbus.

Bits 3 - 0: (0/0/0) Module switches: Bits 0, 1, 2, and 3 correspond to ICMSs 0, 1, 2, and 3 respectively.

Bits 7 - 4: (0/*/0) Global switches: Bits 4, 5, 6, and 7 correspond to ICGSs 0, 1, 2, and 3 respectively.

A.2.13 Interprocessor Communication Registers 0 - 4

Name: ICR0 - 4

Addresses	Registers
0xFFFC 0063	0
0xFFFC 0067	1
0xFFFC 006B	2
0xFFFC 006F	3
0xFFFC 0073	4

Description: These are general-purpose read/write registers that can be accessed from either the local bus or the VMEbus. The addresses listed above are the local addresses. See Section A.1 of this Appendix for details on accessing these registers from the VMEbus.

Bits 7 - 0: (0/0/0) Data field.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.14 Interprocessor Communication Register 5

Name: ICR5

Address: 0xFFFC 0077

Description: This register provides the VIC068A version/revision number. The first VIC068A device contains a value of 0xFFFC 00F1. The address listed above is the local address.

Bits 7 - 0: (Version) VIC068A version/revision (read-only).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.15 Interprocessor Communication Register 6

Name: ICR6

Address: 0xFFFC 007B

Description: This register provides local or remote reset and $\overline{\text{HALT}}$. The address listed above is the local address.

Bits 1 - 0: (X/X/X) Reset/$\overline{\text{HALT}}$ status (read-only from VMEbus): These bits provide reset/$\overline{\text{HALT}}$ status of the VIC068A and local resources according to the following table:

Bit 1 0	Reset/$\overline{\text{HALT}}$ Status.
0 1	$\overline{\text{HALT}}$ has been asserted longer than 6 µs by a source other then the VIC068A. These bits may both be reset by the local CPU to indicate local resources are running and operational.
1 0	The VIC068A has performed a local reset function and the VIC068A is not the system controller. These bits may both be reset by the local CPU to indicate local resources are running and operational.
1 1	Indicates that the CPU has just been released from a system reset.
0 0	Local resources are running and operational. This pattern must be written by the local CPU after a reset condition to indicate that local resources are running and operational.

Bits 5-2: (1/1/1) Undefined/Reserved. Bits will read as 1's.

Bit 6: (1/1/1) $\overline{\text{IRESET}}$ and $\overline{\text{HALT}}$ status (read-only from VMEbus): This bit is set upon assertion of $\overline{\text{IRESET}}$, and/or $\overline{\text{HALT}}$. It is set whether $\overline{\text{HALT}}$ is asserted by external sources or by the VIC068A. $\overline{\text{SYSFAIL}}$ is asserted when this bit is set if the $\overline{\text{SYSFAIL}}$ mask bit (ICR7, bit 7) is cleared.

Bit 7: (X/X/X) $\overline{\text{IRESET}}$ status (read-only): On a VMEbus read, this bit indicates that the VIC068A is in a reset state. On a local bus read, this bit is set whenever $\overline{\text{ACFAIL}}$ is asserted.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.16 Interprocessor Communication Register 7

Name: ICR7

Address: 0xFFFC 007F

Description: This register provides semaphores to the five general-purpose interprocessor communication registers (ICR4 - 0). The remaining bits indicate VMEbus master status, generate $\overline{\text{HALT}}$ and $\overline{\text{RESET}}$, and mask $\overline{\text{SYSRESET}}$. The address listed above is the local address.

Bits 4 - 0:
(0/*/0 [0-3])
(0/X/0 [4])
ICR4 - 0 semaphores: These bits provide semaphores to the five interprocessor communication registers ICR4 - 0 respectively. Each bit is set when the corresponding ICR is written. These bits can be read or written from the local bus or the VMEbus.

Bit 5:
(X/X/X)
VMEbus master status (read-only): This bit is set whenever the VIC068A is the VMEbus master, and the VIC068A is asserting $\overline{\text{AS}}$. This bit is not set when the VIC068A is VMEbus master to an idle bus in ROR and BCAP release modes. Bit 7 of the BESR may be used to indicate that the VIC068A is VMEbus master when $\overline{\text{AS}}$ is not asserted.

Bit 6:
(0/0/0)
$\overline{\text{HALT}}$ and $\overline{\text{RESET}}$ control: This bit may be used to assert the $\overline{\text{HALT}}$ and $\overline{\text{RESET}}$ pins via software. Whenever this bit is set, the VIC068A asserts $\overline{\text{HALT}}$ and $\overline{\text{RESET}}$ until this bit is cleared or any reset occurs.

Bit 7:
(0/*/0)
$\overline{\text{SYSFAIL}}$ mask: When set, the VIC068A is prohibited from asserting $\overline{\text{SYSFAIL}}$ in response to bit 6 of ICR6 being set (which, by default, is set after any reset).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.17 VMEbus Interrupt Request Status Register

Name: VIRSR

Address: 0xFFFC 0083

Description: This register provides status and control of the VMEbus interrupts 7 - 1.

Bit 0: (0/0/0) Register enable/disable: This bit provides enabling and disabling for the remainder of this register.

Bits 7 - 1: (0/*/0) VMEbus interrupt switches: Setting any of these bits asserts the VMEbus $\overline{IRQ}$i signals corresponding to the bit positions, if bit 0 is set during the write. These bits are cleared by setting the appropriate bit and clearing bit 0.

A.2.18 VMEbus Interrupt Vector Base Registers 1 - 7

Name: VIVBR

Address	for $\overline{IRQ}$
0xFFFC 0087	1
0xFFFC 008B	2
0xFFFC 008F	3
0xFFFC 0093	4
0xFFFC 0097	5
0xFFFC 009B	6
0xFFFC 009F	7

Description: Provides the status/ID vector for the VMEbus interrupts.

Bits 7 - 0: (1/*/1 [0-3]) (0/*/0 [4-7]) Status/ID Vector: These bits provide the status/ID vector for VMEbus interrupt acknowledge cycles. Address 0xFFFC 0087 corresponds to $\overline{IRQ1}$. These bits are set to a value of 0x0F for global and system resets and are unchanged by internal resets.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.19 Transfer Timeout Register

Name: TTR

Address: 0xFFFC 00A3

Description: Provides control of the local and VMEbus timeout timers.

Bit 0: (0/0/0) — Include VMEbus acquisition: When set, the local bus timer will include waiting for VMEbus acquisition. When clear, the local bus timer will stop and reset when the VMEbus is requested.

Bit 1: (0/0/0) — Arbitration timeout: When set, the VIC068A as VMEbus arbiter has detected a VMEbus arbitration timeout. This is only used when configured as the VMEbus system controller ($\overline{SCON}$ asserted).

Bits 4 - 2: Local bus timeout period: Defines the local bus timeout.

Bit 4	Bit 3	Bit 2	Local Bus Timeout (µs)
0	0	0	4
0	0	1	16
0	1	0	32 (default - state after any reset)
0	1	1	64
1	0	0	128
1	0	1	256
1	1	0	512
1	1	1	Infinite (timer disabled)

Bits 7 - 5: VMEbus timeout period: Defines the VMEbus timeout.

Bit 7	Bit 6	Bit 5	VMEbus Timeout (µs)
0	0	0	4
0	0	1	16
0	1	0	32
0	1	1	64 (default - state after any reset)
1	0	0	128
1	0	1	256
1	1	0	512
1	1	1	Infinite (timer disabled)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.20 Local Bus Timing Register

Name: LBTR

Address: 0xFFFC 00A7

Description: Provides timing control for $\overline{PAS}$ and $\overline{DS}$ signals when the VIC068A is local bus master. In the following descriptions, **n** is the binary value specified in the bit fields, and **T** is one CLK64M clock period. Clock latency may add one additional clock period to these times.

Bits 3 - 0: (0/*/*) Minimum $\overline{PAS}$ asserted time: This field specifies the *minimum* asserted time for the $\overline{PAS}$ signal whenever the VIC068A is the local bus master. The time is specified by (n+2)T. The actual asserted time depends on a number of factors including local and VMEbus acknowledge timing.

Bit 4: (0/*/*) Minimum $\overline{DS}$ deasserted time: This field specifics the *minimum* deasserted time for the $\overline{DS}$ signal whenever the VIC068A is the local bus master. A time of 1T is selected when this bit is clear; 2T is selected when this bit is set.

Bits 7 - 5: (0/*/*) Minimum $\overline{PAS}$ deasserted time: This field specifies the *minimum* deasserted time for the $\overline{PAS}$ signal whenever the VIC068A is the local bus master. The time is specified by (n+1)T.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.21 Block Transfer Definition Register

Name: BTDR

Address: 0xFFFC 00AB

Description: Configures master block transfers (both MOVEM and block transfers with local DMA) for boundary crossings, dual-path, and user-defined address modifiers. Refer to Chapter 10 of Cypress Semiconductor's **VIC068A/VAC068A User's Guide** for more details on implementing these features.

Bit 0: (0/0/0) Dual-path enable: When set, the VIC068A is enabled with the dual-path feature during master block transfers with local DMA. External logic is required when this option is enabled.

Bit 1: (0/0/0) AMSR Enable: When set, the VIC068A will issue the AM codes based in the address modifier source register for block transfers. This bit effects the AM codes for block transfers only.

Bit 2: (0/0/0) Local boundary crossing enable: When this bit is set, it enables local address 256-byte boundary crossings during DMA block transfer operations. External logic is required to increment latched address lines when this option is enabled.

Bit 3: (0/0/0) VME boundary crossing enable: When this bit is set, it enables VMEbus address 256-byte boundary crossings during DMA block transfer operations. External logic is required to increment latched address lines when this option is enabled.

Bit 7 - 4: (1/1/1) Undefined/Reserved. Bits will be read as 1's.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.22 Interface Configuration Register

Name: ICR

Address: 0xFFFC 00AF

Description: Controls various features of the VIC068A including RMCs, deadlock signaling, metastablility delays, and the "turbo" feature.

Bit 0: (X/X/X) $\overline{\text{SCON}}$ value (read-only): Reads the value of the $\overline{\text{SCON}}$ pin. When set, the VIC068A is not the VMEbus system controller. When clear, the VIC068A is the VMEbus system controller.

Bit 1: (0/0/0) Turbo enable: When set, the VIC068A accelerates VMEbus transfers by reducing selected timings by one CLK64M clock period. VMEbus protocols may be violated when the turbo mode is enabled (see Section 12.5 of Cypress Semiconductor's **VIC068A/VAC068A User's Guide**).

Bit 2: (0/0/0) Metastability interval: When set, the VIC068A adds one additional CLK64M clock period of metastability delay on asynchronous inputs (from 3 CLK64M periods to 4).

Bits 4, 3: (0/0/0) Deadlock signaling: These bits configure deadlock signaling. Bit 4 is used to enable the assertion of $\overline{\text{HALT}}$ and $\overline{\text{LBERR}}$ in addition to the $\overline{\text{DEDLK}}$ signal in deadlock situations. If bit 4 is enabled, bit 3 may be used to prevent the assertion of HALT for RMC deadlocks.

Bit 4	Bit 3	Deadlock Signaling
0	X	$\overline{\text{DEDLK}}$ only (default)
1	0	$\overline{\text{HALT}}$, $\overline{\text{LBERR}}$, $\overline{\text{DEDLK}}$
1	1	$\overline{\text{HALT}}$, $\overline{\text{LBERR}}$, $\overline{\text{DEDLK}}$ ($\overline{\text{HALT}}$ is not asserted for RMC cycles)

Bit 5: (0/0/0) RMC control bit 1: When set, the VIC068A will request the VMEbus whenever the $\overline{\text{RMC}}$ is asserted independent of the $\overline{\text{MWB}}$ signal.

Bit 6: (0/0/0) RMC control bit 2: When set, the VMEbus $\overline{\text{AS}}$ is stretched when $\overline{\text{RMC}}$ is asserted for VMEbus transfers.

Bit 7: (0/0/0) RMC control bit 3: When set, the VIC068A qualifies the RMC control bits 1 and 2 with the SIZ1/0 signals. If RMC control bits 1 or 2 are set and the first cycle of the RMC transfer is of byte size, the "set" behaviors are not implemented.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.23 Arbiter/Requester Configuration Register

Name: ARCR

Address: 0xFFFC 00B3

Description: This register provides configuration of the fairness timeout and DRAM refresh features. The VMEbus request level is also configured from this register.

Bits 3 - 0: (0/0/0) Fairness timer enable: The VMEbus fair requester is enabled in this bit field according to the following table:

Bits 3 - 0	Timeout Period/Mode
0,0,0,0	Fairness disabled (default)
1,1,1,1	Timeout disabled
All other patterns	2 µs times number

Bit 4: (0/*/*): DRAM refresh: When set, the VIC068A will perform CAS-before-RAS ($\overline{DS}$ before $\overline{PAS}$) refresh functions.

Bits 6, 5: (1/1/1) VMEbus request level: The VMEbus request level is set according to the following table:

Bit 6	Bit 5	VMEbus Request Level
0	0	BR0
0	1	BR1
1	0	BR2
1	1	BR3 (default)

Bit 7: (0/0/0) Arbitration mode: When set, the VIC068A performs priority VMEbus arbitration. When clear, the VIC068A performs round-robin arbitration. This bit is only relevant when the VIC068A is configured as the VMEbus system controller ($\overline{SCON}$ asserted).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.24 Address Modifier Source Register

Name: AMSR

Address: 0xFFFC 00B7

Description: This register provides the user-definable address modifiers (AM codes) that can be sourced by the VIC068A for VMEbus master cycles, or used in validating AM codes during VMEbus slave cycles.

Bits 5 - 0: (0/0/0) Address modifier code: The AM code that is issued during master cycles or used for qualifying slave cycles. This register is used only when enabled for user-defined AM codes. Otherwise, standard VMEbus AM codes are used.

Bit 6: (0/0/0) AM5 - 3 qualification: When set, the VIC068A uses bits 5 - 3 in qualifying for slave accesses in addition to the address space size information defined by bits 3 and 2 of the SSiCR0s. This bit is over-ridden if bits 3 and 2 of the SSiCR0s are both clear.

Bit 7: (0/0/0) AM2 - 0 generation: When set, the VIC068A issues the AM2 - 0 codes based on the FC2/1 signals. AM5 - 3 will be issued from bits 5 - 3 of this register. This bit is overridden if the VIC068A is configured to issue all AM bits from this register (i. e., ASIZ1 = ASIZ0 = Low).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.25 Bus Error Status Register

Name: BESR

Address: 0xFFFC 00BB

Description: This register provides $\overline{\text{BERR}}$/$\overline{\text{LBERR}}$, self-access, VMEbus mastership, and timeout status. All bits except bit 7 are flags that must be cleared manually by the local processor after being set by status conditions. If these bits are to be used for a specific operation, it is important that they be cleared prior to starting that operation.

Bit 0: (0/0/0) Local timeout during VMEbus acquisition: This bit, when set, indicates that a local bus timeout has occurred during an attempted acquisition of the VMEbus.

Bit 1: (0/0/0) $\overline{\text{SLSEL1}}$ self-access: This bit is set when the VIC068A is selected by the assertion on the $\overline{\text{SLSEL1}}$ signal, while operating as VMEbus master.

Bit 2: (0/0/0) $\overline{\text{SLSEL0}}$ self-access: This bit is set when the VIC068A is selected by the assertion on the $\overline{\text{SLSEL0}}$ signal, while operating as VMEbus master.

Bit 3: (0/0/0) Local bus timeout: This bit, when set, indicates a local bus timeout occurred without qualification.

Bit 4: (0/0/0) VMEbus timeout: This bit, when set, indicates the VIC068A has signaled a VMEbus timeout. This bit is relevant only if the VIC068A is system controller and the VMEbus timeout is enabled.

Bit 5: (0/0/0) VMEbus bus error: This bit is set when a VMEbus bus error is signaled ($\overline{\text{BERR}}$ asserted).

Bit 6: (0/0/0) Local bus error: This bit is set when a local bus error is signaled a source other then the VIC068A ($\overline{\text{LBERR}}$ asserted to the VIC068A).

Bit 7: (X/X/X) VMEbus mastership: This bit is set whenever the VIC068A is VMEbus master.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.26 DMA Status Register

Name: DMASR

Address: 0xFFFC 00BF

Description: This register provides status of a VIC068A DMA transfer. This includes the block transfer with local DMA function and the module-based DMA function. Status bits are included to show various $\overline{\text{BERR}}$ and $\overline{\text{LBERR}}$ statuses and DMA termination statuses.

Bit 0: (0/0/0) Block transfer in progress. This bit, when set, indicates an interleaved block transfer is in progress. Once set, this bit must be cleared manually by writing a 0 (zero) to this bit location, or by resetting the VIC068A.

Bit 1: (0/0/0) $\overline{\text{LBERR}}$ during DMA transfer: This bit, when set, indicates a $\overline{\text{LBERR}}$ was signaled during a DMA transfer. Once set, this bit must be cleared manually by writing a 0 (zero) to this bit location, or by resetting the VIC068A.

Bit 2: (0/0/0) $\overline{\text{BERR}}$ during DMA transfer: This bit, when set, indicates a $\overline{\text{BERR}}$ was signaled during a DMA transfer. Once set, this bit must be cleared manually by writing a 0 (zero) to this bit location, or by resetting the VIC068A.

Bit 3: (0/0/0) Local bus error (read-only): This bit is set when a local bus error is signaled by a source other then the VIC068A ($\overline{\text{LBERR}}$ asserted to the VIC068A). This bit is a read-only copy of bit 6 of the BESR.

Bit 4: (0/0/0) VMEbus bus error: This bit is set when a VMEbus bus error is signaled ($\overline{\text{BERR}}$ asserted). This bit is a copy of bit 5 of the BESR.

Bits 5, 6: (1/1/1) Undefined/Reserved. These bits will be read as 1's.

Bit 7: (0/0/0) Master write post information stored: This bit is set whenever master write post information is stored.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.27 Slave Select 0 Control Register 0

Name: SS0CR0

Address: 0xFFFC 00C3

Description: This register provides control of the slave selection 0 facilities of the VIC068A. Enabling of the $\overline{\text{LIRQ2}}$ timer interrupt is also configured in this register.

Bits 1 - 0: (0/*/*) Local transfer mode: These bits set the local transfer mode when the VIC068A is local bus master for both slave and master block transfers.

Bit 1	Bit 0	Mode
0	0	No support is given for slave block transfers on $\overline{\text{SLSEL0}}$. The VIC068A will $\overline{\text{BERR}}$ any attempt to receive a VMEbus block transfer. Master block transfers with local DMA will not function in this mode.
0	1	Emulate single-cycle transfers on the local bus. In this mode, the VIC068A emulates single-cycle transfers when performing slave block transfers and master block transfers with local DMA. By emulating single-cycle transfers, the VIC068A toggles the $\overline{\text{PAS}}$ for each cycle. $\overline{\text{DSACKi}}$ must toggle for each transfer and not be held asserted.
1	0	Accelerated transfers on the local bus. In this mode, the VIC068A asserts the $\overline{\text{PAS}}$ signal for the entire slave block transfer and master block transfer with local DMA. The $\overline{\text{DSACKi}}$ signals should be held asserted in this mode.
1	1	Undefined/Reserved.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.27 Slave Select 0 Control Register 0 (continued)

Bits 3 - 2: (0/*/*) Address space configuration: The $\overline{\text{SLSEL0}}$ address space is configured according to the following table:

Bit 3	Bit 2	Address Space
0	0	A32 (extended) (default)
0	1	A24 (standard)
1	0	A16 (short)
1	1	User defined, uses AMSR

Bit 4: (0/*/*) D32 enable: D32 slave operations are enabled for $\overline{\text{SLSEL0}}$ when this bit is set. This bit has no effect for enabling D32 master accesses. This bit also controls byte-lane switching for D16 Block transfers. When set $\overline{\text{ISOBE}}$ and $\overline{\text{SWDEN}}$ alternate states thus alternating which D16 bus data is placed. When clear, only $\overline{\text{SWDEN}}$ is asserted for D16 block transfers.

Bit 5: (0/*/*) Supervisory access: When set, $\overline{\text{SLSEL0}}$ slave accesses are restricted to supervisory accesses. Other accesses are $\overline{\text{BERR}}$ed. Supervisory accesses are checked with the AM(2) signal.

Bits 7-6: (0/0/0) Periodic interrupt timer enable: These bits enable and determine the frequency of the periodic $\overline{\text{LIRQ2}}$ interrupt. If the VIC068A is to handle this local interrupt, LICR2 must be enabled. The frequencies for this interrupt are given below:

Bit 7	Bit 6	Timer Mode
0	0	Timer disabled (default)
0	1	50-Hz output on $\overline{\text{LIRQ2}}$
1	0	1000-Hz output on $\overline{\text{LIRQ2}}$
1	1	100-Hz output on $\overline{\text{LIRQ2}}$

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.28 Slave Select 0 Control Register 1

Name: SS0CR1

Address: 0xFFFC 00C7

Description: This register provides the various access and acquisition timings for slave transfers and slave block transfers for $\overline{\text{SLSEL0}}$ in addition to data acquisition timing for master block transfers with local DMA.

Bits 3 - 0: (0/*/*) Timing field 0: This bit field establishes the following data access/acquisition timings:

- single-cycle slave access timing for $\overline{\text{SLSEL0}}$ (SAT)
- first cycle of a slave block transfer for $\overline{\text{SLSEL0}}$ (SBAT0)
- first cycle of a master block transfer with local DMA (MBAT0)
- first cycle of a module-based DMA transfer (DMAAT0)

The delays are programmed in multiples of the CLK64M clock period according to the table below.

Bit 3	Bit 2	Bit 1	Bit 0	Delay
0	0	0	0	0
0	0	0	1	2.0
0	0	1	0	2.5
0	0	1	1	3.0
0	1	0	0	3.5
0	1	0	1	4.0
0	1	1	0	4.5
0	1	1	1	5.0
1	0	0	0	5.5
1	0	0	1	6.0
1	0	1	0	6.5
1	0	1	1	7.0
1	1	0	0	7.5
1	1	0	1	8.0
1	1	1	0	8.5
1	1	1	1	9.0

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.28 Slave Select 0 Control Register 1 (continued)

Bits 7 - 4: (0/*/*)

Timing Field 1: This bit field establishes the following data access/acquisition timings:

- second and subsequent cycle of a slave block transfer for $\overline{\text{SLSEL0}}$ (SBAT1)
- second and subsequent cycle of a master block transfer with local DMA (MBAT1)
- second and subsequent cycle of a module-based DMA transfer (DMAAT1)

The delays are programmed in multiples of the CLK64M clock period according to the table below.

Bit 7	Bit 6	Bit 5	Bit 4	Delay
0	0	0	0	0
0	0	0	1	2.0
0	0	1	0	2.5
0	0	1	1	3.0
0	1	0	0	3.5
0	1	0	1	4.0
0	1	1	0	4.5
0	1	1	1	5.0
1	0	0	0	5.5
1	0	0	1	6.0
1	0	1	0	6.5
1	0	1	1	7.0
1	1	0	0	7.5
1	1	0	1	8.0
1	1	1	0	8.5
1	1	1	1	9.0

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.29 Slave Select 1 Control Register 0

Name: SS1CR0

Address: 0xFFC 00CB

Description: This register provides control of the slave selection 1 facilities of the VIC068A. Master and slave write posting is enabled in this register as well.

Bits 1 - 0: (0/*/*) Local Transfer Mode: These bits set the local transfer mode when the VIC068A is local bus master for both slave and master block transfers.

Bit 1	Bit 0	Mode
0	0	No support is given for slave block transfers on $\overline{\text{SLSEL1}}$. The VIC068A will $\overline{\text{BERR}}$ any atteinpt to receive a VMEbus block transfer.
0	1	Emulate single-cycle transfers on the local bus. In this mode, the VIC068A emulates single-cycle transfers when performing slave block transfers. By emulating single-cycle transfers, the VIC068A toggles $\overline{\text{PAS}}$ for each cycle. $\overline{\text{DSACKi}}$ must toggle for each transfer and not be held asserted.
1	0	Accelerate transfers on the local bus. In this mode, the VIC068A asserts $\overline{\text{PAS}}$ for the entire slave block transfer. The $\overline{\text{DSACKi}}$ signals should be held asserted in this mode.
1	1	Undefined/Reserved.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.29 Slave Select 1 Control Register 0 (continued)

Bits 3 - 2: (0/*/*) Address Space Configuration: The SLSEL1 address space is configured according to the following table:

Bit 3	Bit 2	Address Space
0	0	A32 (extended) (default)
0	1	A24 (standard)
1	0	A16 (short)
1	1	User defined, uses AMSR

Bit 4: (0/*/*) D32 enable: D32 slave operations are enabled for $\overline{\text{SLSEL1}}$ when this bit is set. This bit has no effect for enabling D32 master accesses.

Bit 5: (0/*/*) Supervisory access: When set, $\overline{\text{SLSEL1}}$ slave accesses are restricted to supervisory accesses. Other accesses are $\overline{\text{BERR}}$ed. Supervisory accesses are checked with the AM(2) signal.

Bit 6: (0/0/0) Master write post enable: When set, master write posting is enabled.

Bit 7: (0/0/0) Slave write post enable: When set, slave write posting is enabled.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.30 Slave Select 1 Control Register 1

Name: SS1CR1

Address: 0xFFFC 00CF

Description: This register provides the various access and acquisition timings for slave transfers and slave block transfers for $\overline{\text{SLSEL1}}$.

Bits 3 - 0: (0/*/*) Timing field 0: This bit field establishes the following data access/acquisition timings:

- single-cycle slave access timing for $\overline{\text{SLSEL1}}$ (SAT)
- first cycle of a slave block transfer for $\overline{\text{SLSEL1}}$ (SBAT0)

The delays are programmed in multiples of the CLK64M clock period according to the table at the bottom of this page.

Bits 7 - 4: (0/*/*) Timing field 1: This bit field establishes the second and subsequent cycle of a slave block transfer for $\overline{\text{SLSEL1}}$ (SBAT1)

The delays are programmed in multiples of the CLK64M clock period according to the table below.

Timing Field 0: Bit 3	Bit 2	Bit 1	Bit 0	
Timing Field 1: Bit 7	Bit 6	Bit 5	Bit 4	Delay
0	0	0	0	0
0	0	0	1	2.0
0	0	1	0	2.5
0	0	1	1	3.0
0	1	0	0	3.5
0	1	0	1	4.0
0	1	1	0	4.5
0	1	1	1	5.0
1	0	0	0	5.5
1	0	0	1	6.0
1	0	1	0	6.5
1	0	1	1	7.0
1	1	0	0	7.5
1	1	0	1	8.0
1	1	1	0	8.5
1	1	1	1	9.0

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.31 Release Control Register

Name: RCR

Address: 0xFFFC 00D3

Description: This register configures the VMEbus release mode. The burst count for block transfers with local DMA is also configured in the RCR.

Bits 5 - 0: (0/0/0) Block transfer burst length: The burst length for both MOVEM block transfers and block transfers with local DMA are configured in this bit field. The value indicates the number of cycles per block transfer (not the number of bytes). A value of 0 in this bit field indicates the maximum 64 cycles per burst. All other values correspond directly to the burst count.

Bits 7, 6: (0/0/0) Release mode: This bit field defines the release mode used by the VIC068A when releasing the VMEbus after the completion of a VMEbus transfer.

Bit 7	Bit 6	Release Mode
0	0	ROR - Release on Request (default)
0	1	RWD - Release When Done
1	0	ROC - Release on $\overline{\text{BCLR}}$ assertion
1	1	BCAP - VMEbus Capture and Hold

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.32 Block Transfer Control Register

Name: BTCR

Address: 0xFFFCC 00D7

Description: The BTCR provides control of the VIC068A block transfers. The local interleave periods and data direction are defined in this register. The enabling bits for all of the VIC068A's block transfer modes are located here as well. These enabling bits are mutually exclusive and more than one should not be set at the same time.

Bits 3 - 0: (0/0/0) Interleave period: The interleave period for block transfers is defined here. The interleave period is 250 ns times the value programmed in this bit field.

Bit 4: (0/0/0) Data direction: This bit defines the direction of a block transfer with local DMA (MOVEM data direction determined by the R/$\overline{W}$ signal). When set, VMEbus block reads occur. When clear, VMEbus block writes occur.

Bit 5: (0/0/0) MOVEM enable: When set, MOVEM transfers are enabled. After this bit is set, the next VMEbus transfer is treated as the start of a VMEbus block transfer. Clearing this bit concludes a MOVEM block transfer in progress. It is important to set this bit immediately before and clear this bit immediately after the actual MOVEM transfer.

Bit 6: (0/0/0) Block transfer with local DMA enable: When set, block transfers with local DMA are enabled. After this bit is set, the next assertion of $\overline{MWB}$ is considered the initiation cycle of a VMEbus block transfer with local DMA. It is important to set this bit immediately before and clear this bit immediately after the actual block transfer.

Bit 7: (0/0/0) Module-based DMA transfer enable: When set, module-based DMA transfers are enabled. After this bit is set, the next assertion of $\overline{BLT}$ by external logic is considered the initiation cycle of a module-based DMA transfer. Clearing this bit concludes a module-based DMA transfer in progress. It is important to set this bit immediately before and clear this bit immediately after the actual DMA transfer.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.2 VIC068A Register Map and Descriptions (continued)

A.2.33 Block Transfer Length Registers 1 - 0

Name: BTLR1 - 0

Addresses: 0xFFFC 00DB (BTLR0), 0xFFFC 00DF (BTLR1)

Description: These registers configure the byte count for block transfers with local DMA. BTLR1 is considered the most significant byte and BTLR0 the least significant. Bit 0 of BTLR0 must never be set because this implies at least one 8-bit transfer is required to complete the block transfer, Only D16 and D32 block transfers are supported. If bit 0 of BTLR0 is set, the block transfer length is ignored and only one burst is performed.

Bits 7 - 0: (0/0/0) Block transfer length: Defines the block transfer length in bytes. BTLR1 contains the most significant 8 bits of the length, and BTLR0 the least.

A.2.34 System Reset Register

Name: SRR

Address: 0xFFFC 00E3

Description: The system reset register provides the means to perform a VMEbus system reset ($\overline{\text{SYSRESET}}$ asserted). Writing a value of 0xFFFC 00F0 causes this function to occur. A system reset is also performed within the VIC068A.

Bits 7 - 0: (1/1/1) System reset field: Writing this bit field with a value of 0xF0 causes $\overline{\text{SYSRESET}}$ to be asserted for a minimum of 200 ms and a system reset to be performed within the VIC068A.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions

The base address for the VAC068A register set is 0xFFFD 0000. Register size is up to 16 bits wide and accesses are acknowledged by using $\overline{\text{DSACK1}}$. The 16-bit registers are **NOT** byte accessible. For single-byte registers, the unused bits are read as 1's. The register names and addresses are summarized in Table A-4, on the next page.

All VAC068A registers are cleared during a global reset and remain intact during a soft reset. Only interrupts are masked during a soft reset. Unused or reserved bits may read as a 0 or a 1. The VAC068A ID register remains intact through all resets.

The VAC068A registers are accessed from the local address/data signals and acknowledged as a 16-bit access by $\overline{\text{DSACK1}}$ assertion. They may only be accessed as a 16-bit word. $\overline{\text{CACHINH}}$ is asserted during accesses to the VAC068A registers. The VAC068A Identification register must be written after reset to enable VAC068A operation.

A.3.1 $\overline{\text{SLSEL1}}$ Address Mask Register

Local Address: 0xFFFD 0000

Bits 31:16 A set bit in any of the positions enables a comparison of the correspondingly numbered local and VMEbus address bits for the purpose of asserting $\overline{\text{SLSEL1}}$.

A.3.2 $\overline{\text{SLSEL1}}$ Base Address Register

Local Address: 0xFFFD 0100

Bits 31:16 The contents of this register are compared under a bitwise mask compare to both the local and VMEbus address bits for the purpose of asserting $\overline{\text{SLSEL1}}$.

A.3.3 $\overline{\text{SLSEL0}}$ Address Mask Register

Local Address: 0xFFFD 0200

Bits 31:16 A set bit in any of the positions enables a comparison of the correspondingly numbered local and VMEbus address bits for the purpose of asserting $\overline{\text{SLSEL0}}$.

A.3.4 $\overline{\text{SLSEL0}}$ Base Address Register

Local Address: 0xFFFD 0300

Bits 31:16 The contents of this register are compared under a bitwise mask compare to both the local and VMEbus address bits for the purpose of asserting $\overline{\text{SLSEL0}}$.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.5 $\overline{\text{ICFSEL}}$ Base Address Register

Local Address: 0xFFFD 0400

Bits 31:24 The upper half of this register is compared to VMEbus address [15:8] for the purpose of asserting $\overline{\text{ICFSEL}}$. When a match occurs between register bits [31:24] and the VMEbus address signals A[15:8], $\overline{\text{ICFSEL}}$ is asserted.

Bits 23:16 The lower half of this register is compared to VMEbus address [15:8] for the purpose of asserting $\overline{\text{ICFSEL}}$. When a match occurs between register bits [23:16] and the VMEbus address signals A[15:8], $\overline{\text{ICFSEL}}$ is asserted.

When either bits A[31:24] or A[23:16] match the VMEbus address bits A[15:8], $\overline{\text{ICFSEL}}$ is asserted. These two different 8-bit compares are used for asserting $\overline{\text{ICFSEL}}$ to the VIC068A for differentiating between module-based or global functions.

A.3.6 DRAM Upper-Limit Mask Register

Local Address: 0xFFFD 0500

This register contains an address mask used to specify the upper address limit of the DRAM memory area located in region 0. The local address bits LA[31:16] are logically ANDed with the NOT of their respective bits in this register and, if all AND outputs are Low, $\overline{\text{DRAMCS}}$ is asserted. A simpler way to view this is if any 1 is present on the address bus where a corresponding 0 exists in the $\overline{\text{DRAMCS}}$ Upper-Limit Mask register, $\overline{\text{DRAMCS}}$ is not asserted.

The logic function used for this compare is a masking operation rather than a full magnitude compare. It is advised that only exact binary multiples be specified for the DRAM memory size (i.e., 1 Mbyte, 2 Mbytes, 4 Mbytes, etc.) to avoid having holes in the local address map. While it is not necessary to fully populate a defined area, $\overline{\text{DRAMCS}}$ will be asserted if an access is attempted to the area, even if there is no memory present.

This register is cleared up on power-up and on global resets. Following exit from the force EPROM mode, accesses to 0x0000 0000 will assert $\overline{\text{DRAMCS}}$ even if no value has been loaded into the mask register.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Table A-4: VAC068A - Register List

Local Address	Register Name	Size
0xFFFD 0000	$\overline{\text{SLSEL1}}$ Address Mask Register	16 bits
0xFFFD 0100	$\overline{\text{SLSEL1}}$ Base Address Register	16 bits
0xFFFD 0200	$\overline{\text{SLSEL0}}$ Base Address Register	16 bits
0xFFFD 0300	$\overline{\text{SLSEL0}}$ Address Mask Register	16 bits
0xFFFD 0400	$\overline{\text{ICFSEL}}$ Base Address Register	16 bits
0xFFFD 0500	DRAM Upper-Limit Mask Register	16 bits
0xFFFD 0600	Boundary 2 Address Register	16 bits
0xFFFD 0700	Boundary 3 Address Register	16 bits
0xFFFD 0800	A24 Base Address Register	13 bits
0xFFFD 0900	Region 1 Attribute Register	6 bits
0xFFFD 0A00	Region 2 Attribute Register	6 bits
0xFFFD 0B00	Region 3 Attribute Register	6 bits
0xFFFD 0C00	$\overline{\text{IOSEL4}}$ DSACK Control Register	16 bits
0xFFFD 0D00	$\overline{\text{IOSEL5}}$ DSACK Control Register	16 bits
0xFFFD 0E00	$\overline{\text{SHRCS}}$ DSACK Control Register	16 bits
0xFFFD 0F00	$\overline{\text{EPROMCS}}$ DSACK Control Register	16 bits
0xFFFD 1000	$\overline{\text{IOSEL0}}$ DSACK Control Register	16 bits
0xFFFD 1100	$\overline{\text{IOSEL1}}$ DSACK Control Register	16 bits
0xFFFD 1200	$\overline{\text{IOSEL2}}$ DSACK Control Register	16 bits
0xFFFD 1300	$\overline{\text{IOSEL3}}$ DSACK Control Register	16 bits
0xFFFD 1400	Decode Control Register	16 bits
0xFFFD 1500	Interrupt Status Register	8 bits
0xFFFD 1600	Interrupt Control Register	16 bits
0xFFFD 1700	Device Location Register	6 bits
0xFFFD 1800	PIO Data Out Register	14 bits
0xFFFD 1900	PIO Pin Register	14 bits
0xFFFD 1A00	PIO Direction Register	15 bits
0xFFFD 1B00	PIO Function Register	16 bits
0xFFFD 1C00	CPU Clock Divisor Register	8 bits
0xFFFD 1D00	UART Channel A Mode Register	12 bits
0xFFFD 1E00	UART Channel A Transmit Data Register	8 bits
0xFFFD 1F00	UART Channel B Mode Register	12 bits
0xFFFD 2000	UART Channel A Receiver FIFO	11 bits
0xFFFD 2100	UART Channel B Receiver FIFO	11 bits
0xFFFD 2200	UART Channel B Transmit Register	8 bits
0xFFFD 2300	UART Channel A Interrupt Mask Register	6 bits
0xFFFD 2400	UART Channel B Interrupt Mask Register	6 bits
0xFFFD 2500	UART Channel A Interrupt Status Register	8 bits
0xFFFD 2600	UART Channel B Interrupt Status Register	8 bits
0xFFFD 2700	Timer Data Register	16 bits
0xFFFD 2800	Timer Control Register	8 bits
0xFFFD 2900	VAC068A ID Register	16 bits

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.7 Boundary 2 Address Register

Local Address: 0xFFFD 0600

This register contains the lower address limit for region 2 and the upper address limit for region 1. Its contents are compared to local address bits LA[31:16]. If the address is less than the value of this register, and neither DRAM nor A24 space (configured in the A24 Base Address register) access occurs, the access is valid for region 1.

A.3.8 Boundary 3 Address Register

Local Address: 0xFFFD 0700

This register contains the upper address limit for region 2 and the lower limit for region 3. The upper address limit for region 3 is the EPROM address space (0xFF00 0000). Its contents are compared to the local address bits LA[31:16]. If the address is less than the value in this register and neither DRAM or A24 space is being accessed, the access is valid for region 2. If the address is greater than or equal to the value in this register yet less than EPROM space, a region 3 access occurs.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.9 A24 Base Address Register

Local Address: 0xFFFD 0800

Bits 31:25 These are compared to local address bits LA[31:25] for the purpose of overlaying an A24 address space in any one of the three regions described by their respective boundaries. Local access to this address space forces a master access to VMEbus A24 space. The area specified must be above the DRAM upper limit.

NOTE: Valid values for bits [31:25] are greater than 0x02 and less than 0xFE.

Bit 24 This bit is decoded along with bit 20 to determine the A24 data path size for the entire 32-Mbyte range. If bit 24 is cleared, a D16 data path is selected. If bit 24 is set, a D32 data path is selected. This bit is only interpreted if bit 20 is cleared.

Bit 23 When set, this bit selects A24 $\overline{\text{CACHINH}}$ (cache inhibit). When cleared, no $\overline{\text{CACHINH}}$ is asserted for A24 address space accesses.

Bit 22 When set, this bit enables bit 21 to determine the data path size of the A16 address space (region 6). When cleared, this bit enables the local address bit LA16 to decode data path size. If LA16 is High, a D16 data path is enabled ($\overline{\text{WORD}}$ asserted). If LA16 is Low, a D32 data path is enabled ($\overline{\text{WORD}}$ deasserted).

Bit 21 When set, this bit along with bit 22 causes region 6 (VMEbus A16 address space) to have a D32 data path ($\overline{\text{WORD}}$ deasserted). When cleared, this bit causes region 6 to have a D16 data path ($\overline{\text{WORD}}$ asserted).

Bit 20 When set, this bit enables the local address bit LA24 to determine the data path size for A24 master accesses. When LA24 is High, the data path size is D16. When LA24 is Low, the data path is D32. When bit 20 is cleared, bit 24 decodes the data path size for the entire A24 address space.

Bit 19 When set, this bit enables $\overline{\text{CACHINH}}$ on accesses to VIC068A register accesses, VAC068A register accesses, and any of the six IOSEL0-5 local I/O address areas. When cleared, $\overline{\text{CACHINH}}$ is not asserted on access to these address spaces.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.10 Region 1-3 Attribute Registers

Local Addresses: 0xFFFD 0900 Region 1 Attribute register.

0xFFFD 0A00 Region 2 Attribute register.

0xFFFD 0B00 Region 3 Attribute register.

Bit 31 When set, this bit enables $\overline{\text{WORD}}$ to be asserted upon access.

Bit 30 When set, this bit enables ASIZ1 to be driven Low upon access.

Bit 29 When set, this bit enables ASIZ0 to be driven Low upon access.

Bit 28 When set, this bit enables $\overline{\text{CACHINH}}$ to be asserted upon access.

Bits 27:26 Follow this table:

Bit 27	Bit 26	Mode
0	0	Inactive
0	1	Shared resources chip select
1	0	VSB resource chip select
1	1	$\overline{\text{MWB}}$ select (VMEbus request)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.11 DSACKi Control Registers

Local Addresses: 0xFFFD 0C00 - $\overline{\text{IOSEL4}}$ $\overline{\text{DSACK}}$i Control Register
(I/O Select Address = 0xFFF8 0000 to 0xFFF9 FFFF)

0xFFFD 0D00 - $\overline{\text{IOSEL5}}$ $\overline{\text{DSACK}}$i Control Register
(I/O Select Address = 0xFFFA 0000 to 0xFFFB FFFF).

0xFFFD 0E00 - $\overline{\text{SHRCS}}$ $\overline{\text{DSACK}}$i Control register
(I/O Select Address is programmable).

0xFFFD 0F00 - $\overline{\text{EPROMCS}}$ $\overline{\text{DSACK}}$i Control register
(I/O Select Address = 0xFF00 0000 to 0xFFEF FFFF).

0xFFFD 1000 - $\overline{\text{IOSEL0}}$ $\overline{\text{DSACK}}$i Control Register
(I/O Select Address = 0xFFF0 0000 to 0xFFF1 FFFF)

0xFFFD 1100 - $\overline{\text{IOSEL1}}$ $\overline{\text{DSACK}}$i Control Register
(I/O Select Address = 0xFFF2 0000 to 0xFFF3 FFFF)

0xFFFD 1200 - $\overline{\text{IOSEL2}}$ $\overline{\text{DSACK}}$i Control Register
(I/O Select Address = 0xFFF4 0000 to 0xFFF5 FFFF)

0xFFFD 1300 - $\overline{\text{IOSEL3}}$ $\overline{\text{DSACK}}$i Control Register
(I/O Select Address = 0xFFF6 0000 to 0xFFF7 FFFF)

Bits 31:29 These bits determine the delay from $\overline{\text{PAS}}$ assertion to assertion of $\overline{\text{DSACK}}$i in CPUCLK cycles per the following table:

000 = 1 CPUCLK cycle
001 = 2 CPUCLK cycles
⋮
111 = 8 CPUCLK cycles

Bit 28 When set, this bit enables $\overline{\text{DSACK1}}$ on slave accesses. When cleared, $\overline{\text{DSACK1}}$ is inactive.

Bit 27 When set, this bit enables $\overline{\text{DSACK0}}$ on slave accesses. When cleared, $\overline{\text{DSACK0}}$ is inactive.

Bits 26:24 These bits determine the recovery time for $\overline{\text{IOSEL}}$i in integer multiples of the CPUCLK as follows:

000 = 1 CPUCLK cycle
001 = 2 CPUCLK cycles
⋮
111 = 8 CPUCLK cycles

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.11 DSACKi Control Registers (continued)

Bits 26:24 (cont'd) — The VAC068A recovery time (time between assertions of device select outputs) is controlled by two separate timers; one for even-numbered $\overline{\text{IOSEL5-0}}$ device select outputs and one for odd-numbered $\overline{\text{IOSEL5-0}}$ device select outputs. Because of the shared usage of these counters, the user must insure that an $\overline{\text{IOSEL5-0}}$ access to a device that uses the same counter (odd or even $\overline{\text{IOSEL5-0}}$ address) is not allowed to start (not issued by the local processor) until the previous access has been deasserted for the required number of CPUCLK cycles. These counters operate only on $\overline{\text{IOSEL5-0}}$ accesses. It is assumed that accesses to $\overline{\text{EPROMCS}}$ or $\overline{\text{SHRCS}}$ do not require a recovery time and should set the value of these bits of their $\overline{\text{DSACKi}}$ Control register to 000.

Bits 23:22 — These bits determine the assertion delay for $\overline{\text{IORD}}$ from $\overline{\text{PAS}}$ in 1/2 CPUCLK cycles (i. e., 0.5, 1, 1.5, 2).

Bits 21:20 — These bits determine the assertion delay for $\overline{\text{IOWR}}$ from $\overline{\text{PAS}}$ in 1/2 CPUCLK cycles (i. e., 0.5, 1, 1.5, 2).

Bits 19:18 — These bits determine the assertion delay for $\overline{\text{IOSEL5-0}}$ from $\overline{\text{PAS}}$ in 1/2 CPUCLK cycles (i. e., 0.5, 1, 1.5, 2).

Bit 17 — When cleared, the $\overline{\text{IORD}}$ signal is deasserted when $\overline{\text{PAS}}$ is deasserted. When set, $\overline{\text{IORD}}$ is deasserted when the time specified in the $\overline{\text{DSACKi}}$ Assertion Delay (bits 31:29) has elapsed. This is used to provide additional hold time for the peripheral device.

Bit 16 — When cleared, the $\overline{\text{IOWR}}$ signal is deasserted when $\overline{\text{PAS}}$ is deasserted. When set, $\overline{\text{IOWR}}$ is deasserted when the time specified in the $\overline{\text{DSACKi}}$ Assertion Delay (bits 31:29) has elapsed. This used to provide additional hold time for the peripheral device.

If bits 18 and 19 are cleared, $\overline{\text{IOSEL5-0}}$ are always deasserted when $\overline{\text{PAS}}$ is deasserted. To use early cycle end control on read cycles, the data must be latched until captured by the local processor, independent of the deassertion of the control signals. This latching function is provided in the VAC068A for devices located on the ID bus signals ID[15:8].

No odd-numbered I/O device access is allowed to start until the select signal for the previously accessed odd-numbered I/O device has been deasserted for the programmed number of clock cycles. The same is true for even-numbered device selects. Recovery time is metered only for $\overline{\text{IOSEL5-0}}$. All other devices are assumed to not need a recovery time. Accordingly, the recovery time field for $\overline{\text{SHRCS}}$ and $\overline{\text{EPROMCS}}$ should be set to 0.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.12 Decode Control Register

Local Address: 0xFFFD 1400

Bit 31 — When set, assert $\overline{\text{DSACKi}}$ on slave accesses during VIC068A local bus cycles (except DRAM Refresh). When cleared, three-state $\overline{\text{DSACKi}}$ on slave accesses during VIC068A local bus cycles.

Bit 30 — When set, qualify $\overline{\text{DRAMCS}}$ assertion with PAS assertion. When cleared, assert $\overline{\text{DRAMCS}}$ on address space match.

Bits 29:28 — These bits specify what local resource select is asserted on $\overline{\text{SLSEL1}}$ assertion when redirection of $\overline{\text{SLSEL1}}$ is enabled (bit 20).

00 = EPROMCS
01 = VSBSEL
10 = SHRCS
11 = DRAMCS

Bit 27 — When set, compare VMEbus A[31:16] to $\overline{\text{SLSEL1}}$ base address register [31:16]. When cleared, no compare. This is used to allocate $\overline{\text{SLSEL1}}$ in an A32/A24 space.

Bit 26 — When set, compare VMEbus A[15:8] to $\overline{\text{SLSEL1}}$ base address register [31:24]. When cleared, no compare. This is used to allocate $\overline{\text{SLSEL1}}$ in an A16 space.

Bit 25 — When set, qualify $\overline{\text{SLSEL0}}$ decode with VMEbus $\overline{\text{AS}}$. When cleared, no qualification.

Bit 24 — When set, qualify $\overline{\text{SLSEL1}}$ decode with VMEbus $\overline{\text{AS}}$. When cleared, no qualification.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.12 Decode Control Register (continued)

Bit 23	When set, qualify $\overline{\text{ICFSEL}}$ decode with VMEbus $\overline{\text{AS}}$. When cleared, no qualification.
Bit 22	When set, qualify boundary decodes (except DRAM) with $\overline{\text{PAS}}$ or $\overline{\text{DS}}$. When cleared, no qualification.
Bit 21	When set, acknowledge DRAM access as 32-bit port (both $\overline{\text{DSACK0/1}}$ asserted). When cleared, three-state $\overline{\text{DSACK0/1}}$ on $\overline{\text{DRAMCS}}$.
Bit 20	When set, redirect $\overline{\text{SLSEL1}}$ area acccsses on local bus to local resource specified in bits 29:28. When cleared, no redirect for $\overline{\text{SLSEL1}}$.
Bit 19	When set, redirect $\overline{\text{SLSEL0}}$ area accesses on local bus to DRAM. When cleared, no redirect.
Bits 18:17	Assertion delay for $\overline{\text{DSACKi}}$ upon access to DRAM (assertion of $\overline{\text{DRAMCS}}$) in CPU clock cycles per the following table: 00 = 0 CPUCLK cycles 01 = 1 CPUCLK cycles 10 = 2 CPUCLK cycles 11 = 3 CPUCLK cycles
Bit 16	When set, $\overline{\text{FPUCS}}$ is asserted on assertion of $\overline{\text{PAS}}$. When cleared, $\overline{\text{FPUCS}}$ is asserted on CPUCLK.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.13 Interrupt Status Register

Local Address: 0xFFFD 1500

Bit 31	When set, this bit indicates that a PIO9 interrupt is pending.
Bit 30	When set, this bit indicates that a PIO8 interrupt is pending.
Bit 29	When set, this bit indicates that a PIO7 interrupt is pending.
Bit 28	When set, this bit indicates that a PIO4 interrupt is pending.
Bit 27	When set, this bit indicates that a mailbox interrupt is pending.
Bit 26	When set, this bit indicates that a timer interrupt is pending.
Bit 25	When set, this bit indicates that a UART A interrupt is pending.
Bit 24	When set, this bit indicates that a UART B interrupt is pending.

This register is read-only. Bits of this register are cleared by accessing the interrupt control register and clearing the control bits for that particular interrupt.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.14 Interrupt Control Register

Local Address: 0xFFFD 1600

Bits 31:30 These bits specify the mapping of PIO9 interrupt to one of the three signals as detailed in the following table.

Bits 29:28 These bits specify the mapping of PIO8 interrupt to one of the three signals as detailed in the following table.

Bits 27:26 These bits specify the mapping of PIO7 interrupt to one of the three signals as detailed in the following table.

Bits 25:24 These bits specify the mapping of PIO4 interrupt to one of the three signals as detailed in the following table.

Bits 23:22 These bits specify the mapping of the mailbox interrupt to one of the three signals as detailed in the following table.

Bits 21:20 These bits specify the mapping of the UART A interrupt to one of the three signals as detailed in the following table.

Bits 19:18 These bits specify the mapping of the UART B interrupt to one of the three signals as detailed in the following table.

Bits 17:16 These bits specify the mapping of the timer interrupt to one of the three signals as detailed in the following table.

Odd bit	Even bit	Function
0	0	Disabled
0	1	Enable to PIO7
1	0	Enable to PIO10
1	1	Enable to PIO11

Note that each interrupt service routine should clear its interrupt's map bits momentarily in order to clear the interrupt request output. Each interrupt is active Low and edge-triggered except UART A and B, which are event triggered and hold until cleared by clearing the interrupt in the Int Mask register. An interrupt request output is asserted only if a falling edge on an interrupt request input occurs while its map bits are non-zero. PIO9 must be held Low for at least 2.8 ms in order to generate an interrupt request.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.15 Device Location Register

Local Address: 0xFFFD 1700

Bit 21 When set, $\overline{\text{IOSEL5}}$ active on the ID bus.

Bit 20 When set, $\overline{\text{IOSEL4}}$ active on the ID bus.

Bit 19 When set, $\overline{\text{IOSEL3}}$ active on the ID bus.

Bit 18 When set, $\overline{\text{IOSEL2}}$ active on the ID bus.

Bit 17 When set, $\overline{\text{IOSEL1}}$ active on the ID bus.

Bit 16 When set, $\overline{\text{IOSEL0}}$ active on the ID bus.

This register specifies mapping of the input/output select device on the ID bus. If any bit is set, it indicates that the corresponding device is located on ID[15:8]. This allows the VAC068A to control the internal Buffer and Latch on the ID bus when these devices are accessed. $\overline{\text{SWDEN}}$ swaps the data between ID[31:24] and ID[15:8] and $\overline{\text{DDIR}}$ controls the data direction.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.16 PIO Data Out Register

Local Address: 0xFFFD 1800

Bit 29	PIO13 or LD29 signal output value.
Bit 28	PIO12 or LD28 signal output value.
Bit 27	PIO11 or LD27 signal output value.
Bit 26	PIO10 or LD26 signal output value.
Bit 25	PIO9 or LD25 signal output value.
Bit 24	PIO8 or LD24 signal output value.
Bit 23	PIO7 or LD23 signal output value.
Bit 22	PIO6 or LD22 signal output value.
Bit 21	PIO5 or LD21 signal output value.
Bit 20	PIO4 or LD20 signal output value.
Bit 19	PIO3 or LD19 signal output value.
Bit 18	PIO2 or LD18 signal output value.
Bit 17	PIO1 or LD17 signal output value.
Bit 16	PIO0 or LD16 signal output value.

This register is used for writing to the PIO signals [13:0] defined as outputs. PIO[13:0] correspond directly to LD[29:16]. When read, the value in the register is driven onto the local data bus LD[29:16]. When written, the value in the register is driven onto those PIO[13:0] signals defined as outputs in the PIO Function register. To set or clear a single PIO[13:0] bit, the register must be read and a logical AND or OR operation performed on the bit, then the value is written back into the register.

A.3.17 PIO Pin Register

Local Address: 0xFFFD 1900

Bits 29:16	Reflect the status of PIO signals [13:0] respectively (i.e., bit 29 = PIO 13, etc.).

This register is read-only and reflects the instantaneous value on those PIO[13:0] signals configured as inputs. Reading this register takes the logic value at the PIO[13:0] signal and drives it onto the local data bus LD [29:16]. Writing to this register causes a $\overline{\text{DSACK1}}$ assertion and has no effect on the contents of the register.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.18 PIO Direction Register

Local Address: 0xFFFD 1A00

Bit 30 — When set, this bit enables $\overline{\text{FCIACK}}$ assertion upon access to 0xFFFF FF00 independent of the function codes. This is useful for interrupt acknowledge emulation for non-68K processors.

Bits 29:16 — These bits correspond directly to PIO signals [13:0]. When set, the direction of the PIO signals are output from VAC068A. When cleared, the direction of the PIO signals [13:0] are input to VAC068A. Thcsc register bits have no effect if the corresponding PIO Function register bits (0xFFFD 1B00) are set.

A.3.19 PIO Function Register

Local Address: 0xFFFD 1B00

Bit 31 — When set, this bit asserts $\overline{\text{FCIACK}}$ upon access to $\overline{\text{IOSEL5}}$ address space independent of the function codes. Also, access to $\overline{\text{IOSEL4}}$ address space asserts $\overline{\text{FPUCS}}$. When cleared, access to $\overline{\text{IOSEL4}}$ and $\overline{\text{IOSEL5}}$ address space does not affect the $\overline{\text{FCIACK}}$ and $\overline{\text{FPUCS}}$ signals.

Bit 30 — When set, this bit enables the debounce delay associated with PIO9 (i. e., 26.7 ms debounce circuit delay). See PIO9 Debounce delay description (Section 19.8.1 of Cypress Semiconductor's **VIC068A/VAC068A User's Guide**) for further details. When cleared, the debounce delay is disabled.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.19 PIO Function Register (continued)

Bits 29:16 These bits select whether the shared function of the PIO pins are enabled. If set, the signal is always an output and operates with the shared function per the following table. If cleared, the signals operate in the PIO mode.

Bit	General Purpose	Shared Function
29	PIO signal 13	$\overline{\text{IOSEL2}}$ address range $FFF4 0000 select
28	PIO signal 12	Shared resources chip select output
27	PIO signal 11	Interrupt request pin 11 (output)
26	PIO signal 10	Interrupt request pin 10 (output)
25	PIO signal 9	$\overline{\text{IOSEL5}}$ address range 0xFFFA 0000 select
24	PIO signal 8	$\overline{\text{IOSEL4}}$ address range 0xFFF8 0000 select
23	PIO signal 7	Interrupt request pin 7 (output)
22	PIO signal 6	$\overline{\text{IOSEL3}}$ address range 0xFFF6 0000 select
21	PIO signa1 5	I/O write signal
20	PIO signal 4	I/O read signal
19	PIO signal 3	UART B receive data signal
18	PIO signal 2	UART B transmit data signal
17	PIO signal 1	UART A receive data signal
16	PIO signal 0	UART A transmit data signal

Interrupt request functions (bits 27, 26, and 23) are mapped in the Interrupt Control register (0xFFFD 1600).

A.3.20 CPU Clock Divisor Register

Local Address: 0xFFFD 1C00

Bits 31:24 These bits set the 16X baud rate clock of 153.6 kHz for use with the VAC068A UART. This register is loaded into an up-counter that continuously counts from the loaded value to 0xFF and reloads on the next clock. The table below gives examples of some CPU clock frequencies and the respective divisor to generate a baud rate of 9600.

CPU Clock	Register Divisor
16 MHz	105
16.67 MHz	108
20 MHz	131
25 MHz	164
30 MHz	196
33 MHz	216

NOTE: Baud rate = CPUCLK/(Divisor x 16)

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.21 UART Channel A and B Mode Register

Local Addresses: 0xFFFD 1D00 Channel A Mode register.

0xFFFD 1F00 Channel B Mode register.

Bit 31 — When set, parity check and generate are disabled. When cleared, parity generate and check are enabled.

Bit 30 — When set, even parity check and generate are enabled. When cleared, odd parity check and generate are enabled.

Bit 29 — When set, data is set for 8 bits per character. When cleared, 7 data bits per character.

Bit 28:26 — These bits set the baud rate for both the transmitter and receiver. The highest baud rate is derived from the CPU Clock Divisor register. The subsequent baud rates are a division of 2 from the previous baud rate. An example follows:

111 = baud rate of 9600
110 = baud rate of 4800
⋮
000 = baud rate of 75

Bit 25 — When set, allows the character receiver to run. When cleared, the receiver is reset.

Bit 24 — When set, allows the character transmitter to run. When cleared, the transmitter is reset.

Bit 23 — When set, ienables the transmitter. When cleared, the transmitter is disabled.

Bit 22 — When set, enables the receiver. When cleared, the receiver is disabled.

Bit 21 — When set, a continuous break is sent. When cleared, break is disabled.

Bit 20 — When set, this bit enables looping of the transmitter output to the receiver FIFO register. When cleared, looping is disabled.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.22 UART Channel A and B Transmit Data Register

Local Addresses: 0xFFFD 1E00 Channel A Transmit Data register.

0xFFFD 2200 Channel B Transmit Data register.

Bits 31:24 These bits are loaded with data to be transmitted via the $\overline{\text{TXD}}$ output when configured in the PIO Function register and enabled in the UART Mode register.

A.3.23 UART Channel A and B Receiver FIFO Register

Local Addresses: 0xFFFD 2000 Channel A Receiver FIFO register.

0xFFFD 2100 Channel B Receiver FIFO register.

Bit 26 When set, this bit indicates that a break error for this byte was detected; otherwise no break error.

Bit 25 When set, this bit indicates that a frame error for this byte was detected; otherwise no frame error.

Bit 24 When set, this bit indicates that a parity error for this byte was detected; otherwise no parity error.

Bits 23:16 Received characters.

The A and B Receiver FIFO registers are read-only.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.24 UART Channel A and B Interrupt Mask Register

Local Addresses: 0xFFFD 2300 Channel A Interrupt Mask register.

0xFFFD 2400 Channel B Interrupt Mask register.

Bit 31	When set, enable interrupt on single character. When cleared, disable interrupt.
Bit 30	When set, enable interrupt on receiver FIFO full. When cleared, disable interrupt.
Bit 29	When set, enable interrupt on break change. When cleared, disable interrupt.
Bit 28	When set, enable interrupt on overrun, framing, or parity error. When cleared, disable interrupt.
Bit 27	When set, enable interrupt on transmitter ready. When cleared, disable interrupt.
Bit 26	When set, enable interrupt on transmitter empty. When cleared, disable interrupt.

The pending interrupt must be disabled in this register, serviced, and then cleared in the Interrupt Control register.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.25 UART Channel A and B Interrupt Status Register

Local Addresses: 0xFFFD 2500 Channel A Interrupt Status register.

0xFFFD 2600 Channel B Interrupt Status register.

Bit 31	When set, this bit indicates that an interrupt has occurred because a character in the receiver is ready to be read.
Bit 30	When set, this bit indicates that an interrupt has occurred because the receiver FIFO is full.
Bit 29	When set, this bit indicates that an interrupt has occurred because a break change was detected.
Bit 28	When set, this bit indicates that an interrupt has occurred because a parity error was detected.
Bit 27	When set, this bit indicates that an interrupt has occurred because a framing error was detected.
Bit 26	When set, this bit indicates that an interrupt has occurred because an overrun error was detected.
Bit 25	When set, this bit indicates that an interrupt has occurred because the transmitter is ready for another character.
Bit 24	When set, this bit indicates that an interrupt has occurred because the transmitter is empty.

This read-only register contains the interrupt status conditions causing the interrupt generated.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.26 Timer Data Register

Local Address: 0xFFFD 2700

Bits 31:16 This register contains the data for loading the VAC068A internal watchdog timer. This data is loaded into a 16-bit up-counter when RUN/$\overline{\text{LOAD}}$ is Low as well as under control of the reload circuitry when ONCE/$\overline{\text{CONTINUOUS}}$ is Low. When the contents of this register are read, the value of the timer is driven onto the data bus, not the value loaded into the register. The counter clock input is driven from the carry out of the prescale counter.

NOTE: Refer to the Timer Control Register (below) for more information on RUN/$\overline{\text{LOAD}}$ and ONCE/$\overline{\text{CONTINUOUS}}$.

A.3.27 Timer Control Register

Local Address: 0xFFFD 2800

Bit 31 ONCE/$\overline{\text{CONTINUOUS}}$: When cleared, the timer counts continuously and interrupts on terminal count. If this bit is set, the timer counts once and stops.

Bit 30 RUN/$\overline{\text{LOAD}}$: When set, the count is enabled and is dependent on bit 31 for control of count cycles. When cleared, the preset value is loaded and the counter is disabled.

Bits 29:24 Prescale Load Value: These bits are loaded into the prescale counter. Bits 29 through 24 correspond directly to D5 through D0 respectively.

The upper two bits of the prescale output (D6, D7) are tied High and not displayed in the register. The prescale counter carry out clocks the count value loaded into the Timer Data register.

Bits 23:16 Prescaler Value: These bits are read-only. They contain the instantaneous value of the prescale counter. Bits 23 through 16 correspond directly to the prescaler counter output Q7 through Q0 respectively.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

A.3 VAC068A Register Map and Descriptions (continued)

A.3.28 VAC068A Identification Register

Local Address: 0xFFFD 2900

Bits 31:20 Constant: These bits are predefined and cannot be changed. A read or write to this register does not affect these bits.

Bits 19:16 Revision number: These bits contain the chip revision number.

VAC068-FS	1AC0
VAC068A	1AC1

NOTE: After a global reset and the completion of loading all other registers, this register must be written in order for the VAC068A to enable its decode and compare functions.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

This page is intentionally blank

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Appendix B: The 4200/01 Distribution Diskette

B.1 Introduction

The diskette that was enclosed with this manual contains all the example programs that were referred to in the body of this manual, and several others as well. "C" language source code is provided that demonstrates how one would work with Pentek's most popular MIX module products when thay are mounted on a Model 4200 or 4201 MIX Baseboard.

The root directory of the diskette contains an archived file, **`4200.zip`**, and two un-archiving utilities (**`pkunzip.exe`** and **`unzip`**). A **`readme`** file is also included, that tells which unarchiver to use and gives the appropriate command syntax, depending on the type of computer you are using. The **`readme`** file is listed below.

```
To restore directory set on a PC running DOS

      pkunzip -d 4200.zip

To restore directory set on a SUN running UNIX

      unzip -a -x 4200.zip
```

B.2 Contents of 4200.zip

When the unarchiving process has been completed, the directory in which **`4200.zip`** was unarchived will contain nine new directories and a new text file, **`readme.1st`**, in addition to the original archive. Below is a list of the directory names and a description of their contents.

Directory	Description
`\4248`	Sample application to demonstrate data collection from a Pentek Model 4248 32 Channel A/D Converter on a Model 4200/01 MIX baseboard
`\4252`	Sample application to demonstrate data collection and output from a Pentek Model 4252 16 Channel A/D-D/AConverter on a Model 4200/01 MIX baseboard
`\4253`	Sample application to demonstrate data output from a Pentek Model 4253 32 Channel D/A Converter on a Model 4200/01 MIX baseboard
`\blink`	Sample application to blink LED on Model 4200/01 front panel
`\demo`	Sample application to demonstrate Model 4200/01 Standard I/O capabilities

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

B.2 Contents of 4200.zip (continued)

Directory	Description
\dma	Sample application to demonstrate DMA capabilities of Model 4200/01
\include	Assorted header files used by the other sample applications
\rel2.0	Contains the compiled code in the 4200/01's Boot Flash EEPROM, and a compiled program to program the flash EEPROMs, in Motorola S record format. Also contains a subdirectory, **\src**
\rel2.0\src	Source code, etc. for the compiled files in the parent directory
\util	Contains another **readme.1st** text file and four subdirectories, **\pc-bit3**, **\sbus-bt3**, **\vme** and **\vxworks**, each of which contain a loader for the 4200/01 and the host monitor program discussed in Section 2.4.14 of this manual. These will be discussed further in Section B.3.

The content of the **readme.1st** file in the main directory is listed below.

```
Model 4200/01 Demo Disk

This disk contains the necessary tools and examples to load and
run programs on the Model 4200/01 Mix Module Baseboard.  All
examples have been compiled on a Sun workstation running Sun OS
ver 4.1.x using the GNU 'C' compiler toolkit provided by :

               Free Software Foundation, Inc.
               675 Mass Ave,
               Cambridge, MA 02139 USA

Getting Started
---------------

1) Connect an RS-232 Terminal or Serial Port on any PC or work-
   station capable of running terminal emulation software to
   serial port A or Monitor Port.  The Serial Port default com-
   munication parameters should be set to 9600 Baud, No Parity,
   8 Data Bits, and 1 Stop Bit.  Refer to section 2.4 in the
   4200/01 manual for a description of features available over
   the monitor port.

2) Select the desired base address settings.  Refer to section
   2.2.2 in the 4200/01 manual to select Base Address settings.

3) Press Reset on the 4200/01's Front Panel and verify the base
   address settings are displayed along with the PENTEK logo on
   the RS-232 Terminal.
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

B.2 Contents of 4200.zip (continued)

4) Run the Blink and Demo Programs using one of the loaders provided in the UTIL directory.

Upgrading 4200/01 Firmware

This disk contains the executable and source code for Revision 2.0 Firmware for the 4200/01. Firmware can be upgraded two ways:

Serial Port(Revision 1.84 or Later)
VME

Note: Be aware that downloading new firmware to the 4200/01 will erase the unit's existing boot code. User-generated code should NOT be programmed in this manner. If the 4200/01 doesn`t boot after performing the upgrade (LED stays off upon RESET), contact PENTEK at (201)767-7100.

Upgrading Firmware over VME
---

1) Before you try to load the new firmware, verify that programs can be loaded using one of the loaders provided on this disk. They may be different from the one you currently use.

2) Enable the Flash EEPROM to be written by moving the jumper on the JB5 jumper block to position 1-2.

3) Loading the firmware consists of 2 steps:

Loading the Bootcode to SRAM at location Base Address + 0x10000

Loading and Executing a program which erases the Flash EEPROM and writes the new Bootcode. This program is loaded in SRAM at location Base Address + 0x50008.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

B.2 Contents of 4200.zip (continued)

4) Enter the following commands:

```
load42 -l -pxxx1 -myy bootrom.hex
```

Where:

xxx1 - Bits A31-A16 of the base address for the SRAM. Bits A31-A24 only need to be specified for A32 address mode.

yy - Addressing Mode (24 or 32)

The loader may request that a key be pressed after the 4200/01 has completed Resetting.

```
load42 -x -pxxxx -myy progrom.hex
```

This program will be executed after it is loaded. On the Serial Port Terminal you will see the following messages:

```
Erasing Device...Done
Verifying Erase...Done
Now Programming rom...Done
Verifying...Done
```

The 4200/01 will reset and re-display the logo along with the current revision number.

Example:

To load bootcode to a 4200/01 with it's Base Address at 0x80000000 in A32 Addressing Mode.

```
load42 -l -p8001 -m32 bootrom.hex
load42 -x -p8000 -m32 progrom.hex
```

To load bootcode to a 4200/01 with it's Base Address at 0x800000 in A24 Addressing Mode.

```
load42 -l -p81 -m24 bootrom.hex
load42 -x -p80 -m24 progrom.hex
```

5) Replace the Write Protect jumper on JB5 to positions 2-3.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

B.2 Contents of 4200.zip (continued)

```
Upgrading Firmware over Serial Port
-----------------------------------

1) Enable the Flash EEPROM to be written by moving the jumper on
   the JB5 jumper block to position 1-2.

2) Select the Upgrade Firmware Option in the 4200/01 monitor's
   menu.

3) Upload the program selecting ASCII Protocols and the file to
   be uploaded.

Note:   The following parameters for ASCII transfers need to be
        specified in the terminal emulation software being used:

        Carriage Return translation     - None
        Line Feed translation           - None
        Space between sending each line - .2 seconds
        ECHO locally                    - No

4) Upon successful transfer, the Monitor will provide feedback
   on the number of bytes loaded and request whether to Upgrade
   the Firmware or Abort.  If you select the Program option, the
   following will display on the terminal.

        Erasing Device...Done
        Verifying Erase...Done
        Now Programming rom...Done
        Verifying...Done

   The 4200/01 will reset and re-display the logo along with the
   current revision number.

5) Replace the Write Protect jumper on JB5 to positions 2-3.
```

B.3 Utility Programs for the Model 4200/01

The **\util** directory that was created when **4200.zip** was unarchived contains subdirectories whose names describe the various host system connections currently in use among most of Pentek's customer base. Those directories each contain source and executable files for a program to download code to the Model 4200/01 (**load42.c** and **load42.exe**), and for the host monitor program descrbed in Section 2.4.14 of this manual (**mon42.c** and **mon42.exe**). There is also another **readme.1st** text file, to describe the use of these programs. A listing of that files contents begins at the top of the next page.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

B.3 Utility Programs for the Model 4200/01 (continued)

```
        4200/01 Utility Programs

This disk contains a Loader and VME Monitor program for the fol-
lowing Platforms:

        Embedded Sparc or Sun running SUN OS
        Embedded Processor running VxWorks
        Sun with a Bit 3 SBUS to VME Adapter
        PC with a Bit 3 PC to VME Adapter

Load42 Utility
--------------

This program enables the user to download and execute programs
in the 4200/01's SRAM.  Programs must be in the Motorola 'S'
Record Format.

To Invoke:

   load42 -xlmp file

   Arguments:

   -x      Do Not Perform a VME Reset prior to loading program.
           If a reset is performed, a request will be made to
           press any key to start.  This will allow the 4200 to
           finish initializing itself.

   -l      Load the program only.  Do not execute

   -m      Specify addressing mode(24 or 32).

   -p      Specify the upper 8 address bits.  This can also be
           used to offset the program in SRAM.

   file    Executable in Motorola 'S' record format.

To Compile:

For   Sun OS:    cc load42.c -o load42
      PC:        Used Turbo 'C', may need to be modified for
                 other compilers
      VxWorks:   Used GNU Compiler
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

B.3 Utility Programs for the Model 4200/01 (continued)

Monitor Utility

This program enables the user to command the 4200/01's built in monitor to perform certain functions. The available functions are listed in Section 2.4.14 in the 4200/01 Manual.

```
For   Sun OS:    cc mon42.c -o mon42
      PC:        Used Turbo 'C', may need to be modified for
                 other compilers
      VxWorks:   Used GNU Compiler
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

This page is intentionally blank

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Appendix C: MIX Assembly and Installation Instructions

C.1 Introduction

The connection between Pentek's family of expansion modules and the VMEbus re–sources is provided by the Modular Interface eXtension (MIX) interface, originally developed by Intel Corporation for Multibus II. The MIX standard defines an electrical and mechanical protocol for a private high–speed 32–bit bus, fully supported with interrupt and control functions. The elements of the MIX system, as implemented by Pentek for VMEbus, are described in this appendiC.

C.2 A Glossary of MIX Terms

MIX Baseboard — The primary VME–to–MIX interface board designed for supporting one, two, or three MIX expansion modules. A MIX Stack must contain one, and only one, baseboard.

MIX Expansion Module — A printed circuit board that mounts onto the MIX Baseboard. Expansion modules may provide I/O, Memory or processing resources for the baseboard. An expansion module may have a Slave–only or a Master/Slave MIX interface, and may or may not possess its own VMEbus interface. The terms MIX module and expansion module are used interchangably in this appendiC.

VME Board — A self–contained, standalone assembly suitable for plugging into a VME card cage.

MIX Board — The primary Printed Circuit (PC) board within the expansion module (i. e., the board containing the MIX surface mount connector pattern). In some expansion modules, (such as the Pentek Models 4241 and 4244), this is the only PC board.

Mezzanine Board — A secondary PC board attached to the MIX board containing additional circuitry. For example, in the Model 4242, the mezzanine board contains the analog circuitry, including amplifiers, filters, and the A/D and D/A converters.

MIX Bus Master — A device that initiates and controls data transfer cycles over the MIX bus. When this device is the MIX Baseboard, it is referred to as a Lower MIX Bus Master (LMBM). When an expansion module masters a MIX transaction, it is called an Upper MIX Bus Master (UMBM).

MIX bus Slave — A device that participates in, but neither initiates nor controls, a data transaction over the MIX bus.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.3 Pentek MIX Baseboards

Pentek offers a variety of MIX baseboards, all of which are full size VMEbus boards in standard 6U format (160 mm x 233 mm) and occupy one standard slot position (0.8 inches of horizontal space) in a VMEbus card cage. Each baseboard has two 96–pin male DIN connectors (P1 and P2) that mate with the corresponding female connectors on the VMEbus backplane. All power for the baseboard, and any expansion modules it may hold, is drawn from the VMEbus through these connectors.

The MIX baseboard features a surface mount contact pattern near the center of the board that supports the MIX interface. This pattern consists of 130 gold–plated pads on the upper and lower surfaces of the baseboard and mounting holes for the MIX connectors.

Pentek's line of MIX baseboards for the VMEbus includes the following devices:

- Model 4200: 68030–based; 16 MHz processor; VMEbus A32/D32 Master/Slave and VSB Master interfaces; up to 4 MB RAM
- Model 4200A: 68030–based; 40 MHz processor; VMEbus A32/D32 Master/Slave and VSB Master interfaces; up to 4 MB RAM
- Model 4201: 68030–based; 16 MHz processor; VMEbus A32/D32 Master/Slave interface; up to 4 MB RAM
- Model 4201A: 68030–based; 40 MHz processor; VMEbus A32/D32 Master/Slave interface; up to 4 MB RAM
- Model 4202: VMEbus A32/D32 Slave–only Interface; maps MIX bus addresses to VMEbus addresses
- Model 4283: DSP–based (TMS320C30); VMEbus A32/D32 Master/ Slave interface; up to 8 MB RAM
- Model 4284: DSP–based (TMS320C40); VMEbus A32/D32 Master/ Slave interface; up to 16 MB RAM
- Model 4285: DSP–based (Multiple TMS320C40s); VME64 Master/Slave and Auxiliary A32/D32 VMEbus Master interfaces; up to 22 MB RAM

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.4 Pentek MIX Expansion Modules

Pentek currently offers MIX expansion modules for the VME bus with the following features:

- ❑ Floating Point Digital Signal Processors:
  - ♦ Single and Dual Texas Instruments TMS320C30 Processors
  - ♦ Single, Dual, and Quad Texas Instruments TMS320C40 Processors
  - ♦ Triple AT&T DSP32C Processors
- ❑ A/D Converters
  - ♦ 32 Channel, 16–bit, 12.5 kHz sample rate
  - ♦ 32 Channel, 12– or 14–bit, 125 kHz sample rate
  - ♦ 32 Channel, 16–bit, 100 kHz sample rate with digital filter
  - ♦ Single Channel, 12–bit, 1 MHz sample rate with LC filter
  - ♦ Single Channel, 12–bit, 10 MHz sample rate
  - ♦ Single Channel, 14–bit, 10 MHz sample rate with 'C40 Comm Port output
- ❑ A/D – D/A Converters
  - ♦ 16 Channel, 16–bit, 48 kHz sample rate
  - ♦ 8 Channel, 16–bit, 200 kHz sample rate, with 'C40 Comm Port I/O
  - ♦ Single and Dual Channel, 18–bit, 200 kHz sample rate
  - ♦ Single Channel, 12–bit, 1 MHz sample rate
- ❑ D/A Converter: 32 Channel, 12–bit, 100 kHz sample rate
- ❑ Digital I/O devices
  - ♦ 32–bit Parallel Digital I/O
  - ♦ SCSI–1 Interface
  - ♦ Wide/Fast SCSI–2 Interface
  - ♦ RS–232C Serial Interface
- ❑ Digital Receivers
  - ♦ Four Channel, 12–bit, 70 MHz Digital Drop Receiver
  - ♦ Multiband Digital Drop Receiver
- ❑ Single and Dual Channel T1/CEPT Interface
- ❑ Time Code Reader
- ❑ Prototyping Module

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.5 MIX Module General Description

Standard MIX modules are 10.3" long x 4.0" deep x 0.8" wide. No electrical or mechanical connection is made between the VMEbus back plane and a standard MIX module, since all power and signal lines are provided through the MIX interface connectors. Each module has its own front panel secured to the card cage with captive hardware. The back ends of standard expansion modules are mechanically secured to the baseboard by means of the MIX connectors.

Some of Pentek's MIX modules differ from the above standards. The Model 4240 MIX prototyping module is somewhat deeper (6.0"), to allow more flexibility for custom circuit construction.

The Model 4270, with four TMS320C40 DSPs can function as either a stand–alone VME Slave board, a MIX Master/Slave module, or both. This unit is therefore a full–size 6U VME board, and draws power from the VME backplane, rather than from the MIX bus. The Model 6102 Octal A/D–D/A Converter has a full VMEbus slave interface as well as a MIX bus slave interface. This, too, is a full–depth MIX card, which engages the P1 and P2 VME connectors and can participate in VMEbus data cycles. On these full–depth MIX modules, the MIX connector pattern is located near the center of the card, rather than at the rear.

The Emitter–Coupled Logic (ECL) input versions of the Digital Receiver modules (Models 4271 and 4272) are also full depth MIX cards. These also draw power from the VMEbus, but cannot participate in any VME data transactions (although they do pass along the daisy–chained signals).

Each MIX module is accessed by the MIX baseboard as a unique memory region, determined by the module's stack position. Separate interrupt and control lines are provided for each stack position.

C.6 MIX Module Stack Assembly Techniques

Two methods are available for installing the first MIX module on the MIX baseboard. The first expansion module may be mounted in the Nested position or in the Non–Nested position. These terms will be defined below.

When the first expansion module is mounted in the Nested position, this module and the MIX baseboard occupy the same VME card cage slot, and share the expansion module's front panel. When the nested assembly technique is used, a full MIX stack (consisting of a baseboard and three expansion modules) occupies only three slots in the VME card cage. This technique leaves an additional slot in the card cage available, so that another VME device may be installed.

When the first expansion module is mounted in the Non–Nested position, the baseboard occupies its own slot in the VMEbus card cage, and has its own front panel. The first expansion module will occupy the slot to the right of the baseboard in the card cage. When the Non–Nested assembly technique is used, a full MIX stack (consisting of a baseboard and three expansion modules) occupies four slots in the VME card cage.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.6 MIX Module Stack Assembly Techniques (continued)

The front panels of the Models 4200, 4200A, 4201 and 4201A are very useful for the initial configuration of these baseboards, but in normal use the panel is generally removed. These baseboards will normally utilize the Nested assembly technique to share a common card cage slot with the first MIX expansion module.

MIX Baseboard Model 4202 is most often used without the VME front panel, and with the first expansion module mounted in the Nested position. The Model 4283 can sometimes be used in this manner as well. The Models 4284 and 4285 Baseboards do not support the Nested module installation, and each of these devices will always occupy its own card cage slot.

Full–depth MIX modules can not be Nested on any baseboard, because they must engage the VME backplane and therefore require a card cage slot of their own.

Table C–1 (below) describes which combinations of MIX Baseboards and expansion modules allow for Nested installation.

Table C–1: Expansion Module/Baseboard Nesting Compatibility

A 'Yes' indicates that for the given MIX Baseboard/Module combination, nested installation in stack position 1 is possible.

MIX Modules	Pentek VME/MIX Baseboards					MIX Modules	Pentek VME/MIX Baseboards				
	4200(A)	4201(A)	4202	4283	4284/5*		4200(A)	4201(A)	4202	4283	4284/5*
4240	No	No	No	No	No	**4254**	Yes	Yes	Yes	No	No
4241	Yes	Yes	Yes	Yes	No	**4255**	Yes	Yes	Yes	Yes**	No
4242	Yes	Yes	Yes	No	No	**4256**	Yes	Yes	Yes	Yes	No
4243	Yes	Yes	Yes	No	No	**4257**	Yes	Yes	Yes	No	No
4244	Yes	Yes	Yes	Yes	No	**4258**	Yes	Yes	Yes	No	No
4245	Yes	Yes	Yes	No	No	**4259**	Yes	Yes	Yes	No	No
4246	Yes	Yes	Yes	No	No	**4260**	Yes	Yes	Yes	No	No
4247	Yes	Yes	Yes	No	No	**4270**	No	No	No	No	No
4248	Yes	Yes	Yes	No	No	**4271**	Yes†	Yes†	Yes†	No	No
4249	Yes	Yes	Yes	No	No	**4272**	Yes†	Yes†	Yes†	No	No
4250	Yes	Yes	Yes	No	No	**4273**	Yes	Yes	Yes	No	No
4251	Yes	Yes	Yes	No	No	**4274**	Yes	Yes	Yes	No	No
4252	Yes	Yes	Yes	No	No	**4275**	Yes	Yes	Yes	No	No
4253	Yes	Yes	Yes	No	No	**6102**	No	No	No	No	No

* – MIX Baseboard Models 4284 and 4285 do not support nested installation

** – Model 4255, Option 001, which features shrouded, latching connectors, will not nest on a Model 4283 baseboard

† – The ECL Input versions of these devices connect to VME, and cannot be nested on any baseboard

Regardless of the assembly technique selected, any MIX stack may consist of a baseboard with one, two, or three expansion modules installed upon it. The expansion modules are designed such that, when installed, the component side of the module faces the component side of the the Baseboard.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.7 MIX Module Stack Assembly Components

The following subsections describe the mechanical components used to build a MIX stack. Figures C–1 through C–3, on this page and the next, present pictures of these components. Figure C–4, on the page after next, shows how these components are used in the assembly of a MIX stack.

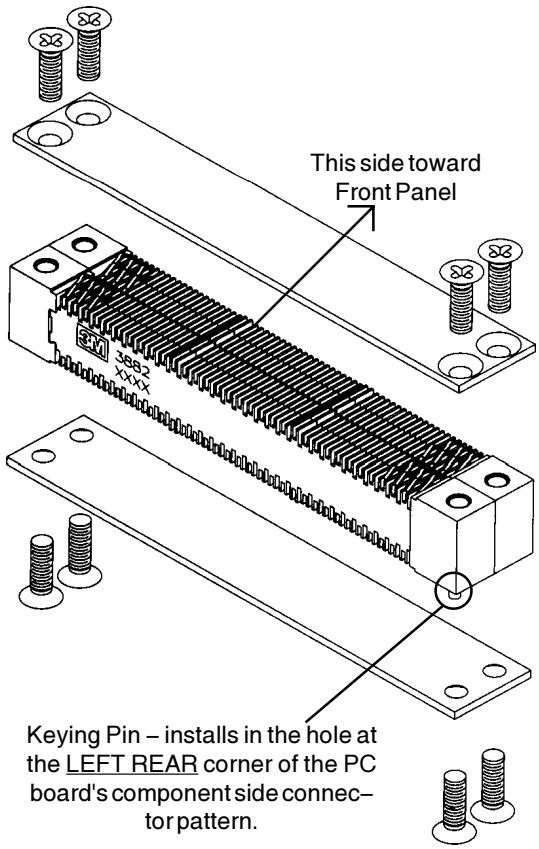


Figure C–1: MIX Baseboard Connector with Stiffeners and Flathead Screws.

Viewed from rear of Baseboard

Manufactured by 3M Electronic Products Division as Part Number X3882–1000, which includes one baseboard connector, two stiffeners and eight screws. This kit is shipped with each Pentek MIX baseboard.

Reproduced with permission of 3M.

C.7.1 MIX Baseboard Connector

The MIX baseboard connector attaches a MIX expansion module to a MIX baseboard. The MIX baseboard connector (see Figure C–1, left) is a surface mount bridge between the MIX contact pattern on the baseboard and the matching contact pattern on an expansion module. The baseboard connector is 0.476" tall and has 130 gold–plated spring contacts on both top and bottom surfaces. It also features a keying pin, that fits into a keying hole on the baseboard. Pentek applies bright, orange labels to all MIX connectors, showing the "UP" side of the connector (i. e., toward the next module position) and the side of the connector that faces the front panels.

C.7.2 MIX Stiffener

The MIX stiffener maintains contact pressure between the MIX baseboard connector and the MIX baseboard. Uninstalled, the MIX stiffener is slightly curved and features an insulating polyester film on the surface that touches the MIX contact pattern. The MIX stiffener attaches to the MIX baseboard connector via four flathead screws which pass through the MIX baseboard and into threaded inserts in the MIX baseboard connector. Figure C–1, left,also shows the top and bottom stiffeners and the flathead screws.

When tightening these flathead screws, the torque should not exceed 2 in–lbs or there is a possibility of damaging the threads in the MIX baseboard connector. Pentek offers a special torque driver specifically for tightening the flathead screws (Pentek Model # 2030).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.7 MIX Module Stack Assembly Components (continued)

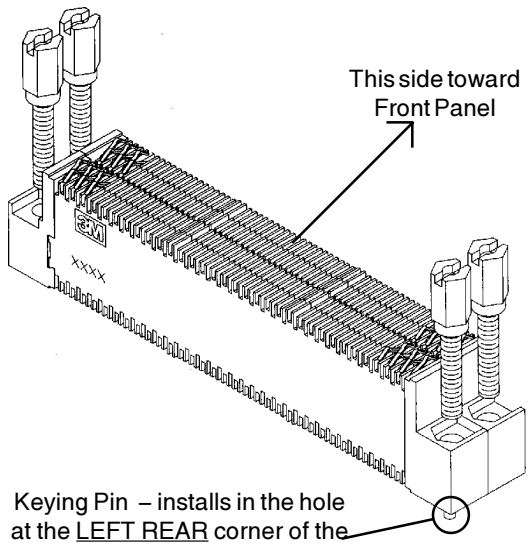


Figure C–2: MIX Stacking Connector with Jackscrews.

Viewed from rear of PC Board

Manufactured by 3M Electronic Products Division as Part Number X3884–0000, which includes one stacking connector and four jackscrews. This kit is shipped with each expansion module.

Reproduced with permission of 3M.

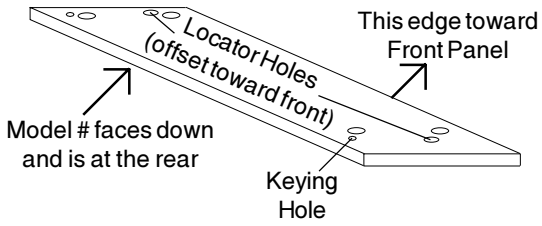


Figure C–3: MIX Spacer Board

Viewed from rear of Baseboard

Manufactured by Tektronix. The 'MIX' spacer board is shipped with every Pentek MIX baseboard.

C.7.3 MIX Stacking Connector

The MIX stacking connector is used when adding expansion modules to the MIX stack. The stacking connector is similar to the MIX baseboard connector described in Section C.7.1, except that its height is 0.738 inches instead of 0.476 inches and the stacking connector features a step at each end to accommodate the jackscrews (described in Section C.7.5. below). The stacking connector is shown in Figure C–2, left.

C.7.4 MIX Spacer Board

In situations where the first MIX expansion module cannot (or will not) be nested on the baseboard (see Table C–1, page before last), a MIX spacer board (see Figure C–3, below left) must be installed between the MIX baseboard connector and the MIX stacking connector. A MIX spacer board is supplied with every Model 4283 and Model 4284 baseboard, and with the non–nestable MIX modules.

C.7.5 MIX Jackscrews

Each stacking connector attaches to the MIX connector below it via MIX jackscrews (see Figure C–2, above left). The jackscrews pass through the MIX stacking connector, through the MIX expansion board, and into the threaded inserts (or other jackscrews) in the MIX connector below. Each jackscrew has a threaded opening in the head to accept jackscrews from additional MIX components above it. Pentek offers a special torque driver specifically for tightening the MIX jackscrews (Pentek Model # 2031).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

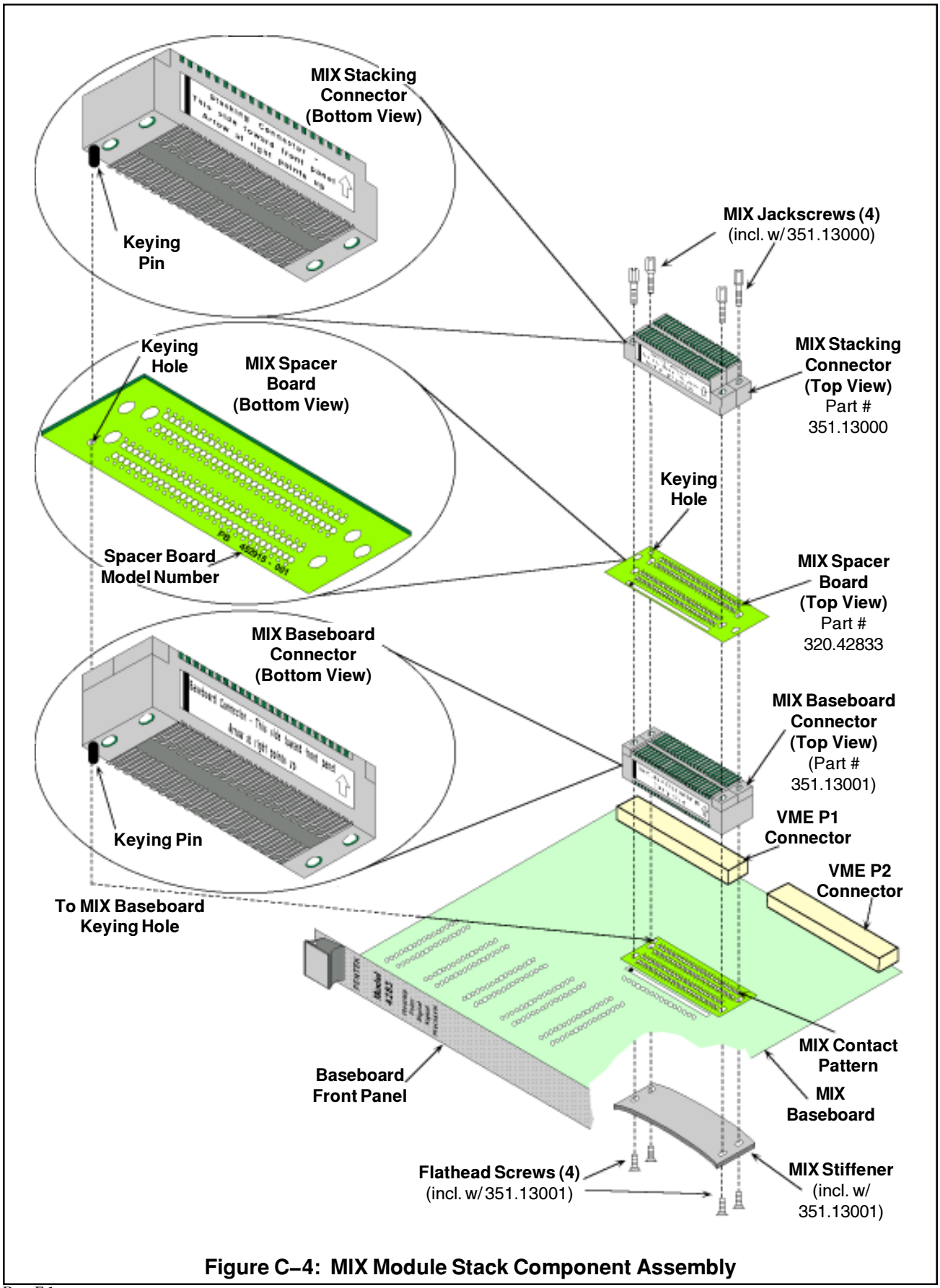


Figure C–4: MIX Module Stack Component Assembly

Rev. F.1

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

This page is intentionally blank

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.8 Non–Nested MIX Stack Assembly

While performing these assembly procedures, refer to Figures C–4, C–5 and C–6, on pages C–8, C–11 and C–13, respectively. Expansion module positions are referred to as "MIX stack position 0" for the first module (the module mounted ***closest to the base–board***), "stack position 1" for the second module, and "stack position 2" for the third module.

Assemble all MIX components in a static controlled environment!



Improper alignment of the MIX components *WILL* cause failure of the MIX interface and very likely damage the boards!

Turn off all power to the card cage before inserting or removing any board(s)!

1) Place a MIX baseboard connector on top of the baseboard's MIX contact pattern. Verify that the black stripes on both orange labels are aligned, and that the baseboard connector's keying pin is seated in the keying hole of the baseboard's contact pattern.

2) Place the insulated (shiny) side of a MIX stiffener against the **BOTTOM** of the baseboard's MIX contact pattern.

3) Insert, but ***do not fully tighten***, four flathead screws through the MIX stiffener, through the baseboard, and begin threading them into the baseboard connector.

4) Tighten the flathead screws ***slowly*** and ***evenly***, one turn at a time for each screw in sequence, following the pattern shown in Figure C–5A, to draw the baseboard connector against the baseboard's MIX contact pattern.

   NOTE: Do not torque the flathead screws over 2 in–lbs (Pentek Model #2030 Torque Driver may be used).

5) Place a MIX spacer board on top of the MIX baseboard connector such that the black stripes on all three orange labels are aligned.

6) Place a MIX stacking connector over the MIX spacer board such that the black stripes on all four orange labels are aligned. Verify that the stacking connector's keying pin seats properly in the spacer board's keying hole.

7) Insert, but ***do not fully tighten***, four jackscrews through the stacking connector, through the spacer board, and begin threading them into the baseboard connector.

8) Tighten the jackscrews ***slowly*** and ***evenly***, one turn at a time for each screw in sequence, following the pattern shown in Figure C–5A, to draw the stacking connec–tor against the spacer board's MIX contact pattern.

   NOTE: Do not torque the jackscrews over 5 in–lbs (Pentek Model #2031 Torque Driver may be used).

9) Place the ***first*** expansion module over the stacking connector, with the component side facing ***DOWN***. ***If this installation requires only one expansion module, proceed to step 18*** (see next page).

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

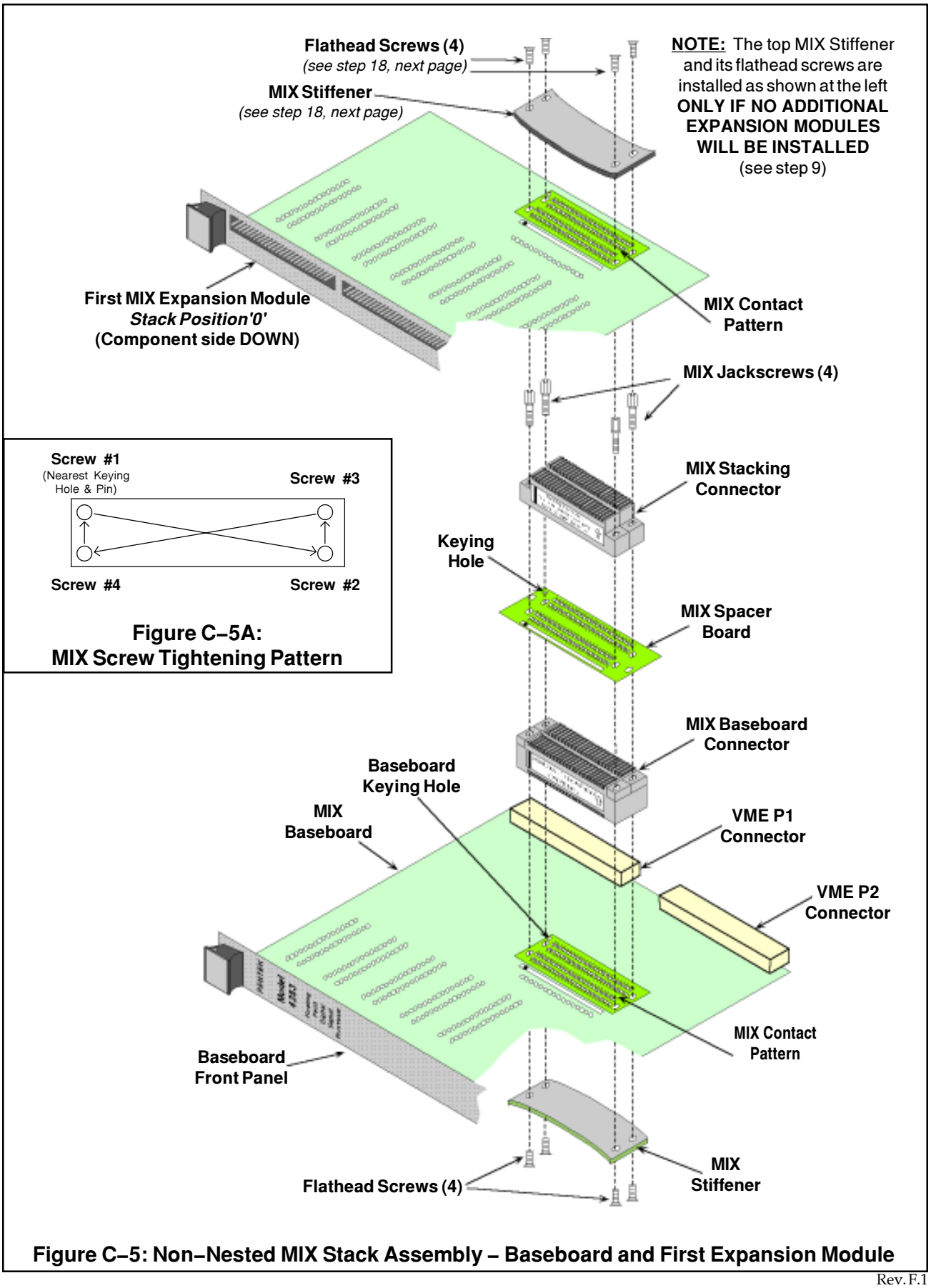


Figure C–5A: MIX Screw Tightening Pattern

Figure C–5: Non–Nested MIX Stack Assembly – Baseboard and First Expansion Module

Rev. F.1

C.8 Non–Nested MIX Stack Assembly (continued)

10) Place another stacking connector over the MIX contact pattern on the TOP of the ***first*** expansion module (see Figure C–6, on the next page). Verify that the black stripes on all orange labels are aligned, and that the stacking connector's keying pin seats properly in the expansion module's keying hole.

11) Insert, but ***do not fully tighten***, four jackscrews through the stacking connector, through the ***first*** expansion module, and begin threading them into the stacking connector below the expansion module.

12) Tighten the jackscrews ***slowly*** and ***evenly***, one turn at a time for each screw in sequence, following the pattern shown in Figure C–5A (on the previous page), to draw the stacking connector against the expansion module.

    NOTE: Do not torque the jackscrews over 5 in–lbs (Pentek Model #2031 Torque Driver may be used).

13) Place the ***second*** expansion module over the stacking connector with the component side facing ***DOWN. If this installation requires only two expansion modules, proceed to step 18***, below.

14) Place another stacking connector over the MIX contact pattern on the TOP of the ***second*** expansion module (see Figure C–6, on the next page). Verify that the black stripes on all orange labels are aligned, and that the stacking connector's keying pin seats properly in the ***second*** expansion module's keying hole.

15) Insert, but ***do not fully tighten***, four jackscrews through the ***third*** stacking connector, through the ***second*** MIX expansion module, and begin threading them into the ***second*** stacking connector.

16) Tighten the jackscrews ***slowly*** and ***evenly***, one turn at a time for each screw in sequence, following the pattern shown in Figure C–5A (on the previous page), to draw the stacking connector against the expansion module.

    NOTE: Do not torque the jackscrews over 5 in–lbs (Pentek Model #2031 Torque Driver may be used).

17) Place the ***third*** expansion module over the stacking connector, with the component side facing ***DOWN***.

18) Place the insulated (shiny) side of the MIX stiffener against the MIX contact pattern on the TOP of the ***last*** expansion module.

19) Insert, but ***do not fully tighten***, four flathead screws through the MIX stiffener, through the ***last*** expansion module, and begin threading them into the stacking connector below.

20) Tighten the flathead screws ***slowly*** and ***evenly***, one turn at a time for each screw in sequence, following the pattern shown in Figure C– 5A (on the previous page) to draw the stacking connector against the ***last*** expansion module's MIX contact pattern.

    NOTE: Do not torque the flathead screws over 2 in–lbs (Pentek Model #2030 Torque Driver may be used).

This completes the assembly procedure for a non–nested MIX stack.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

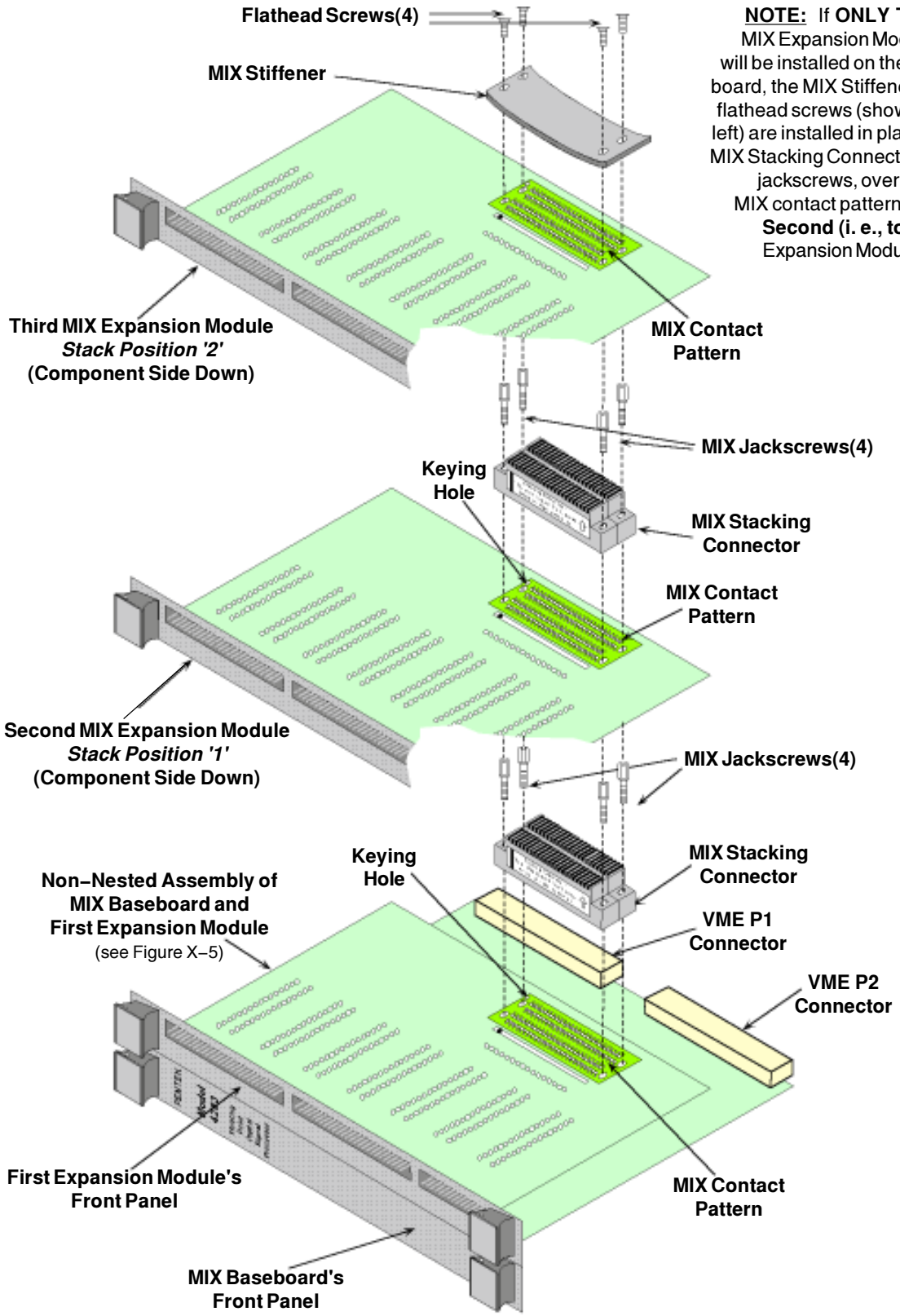


Figure C–6: Non–Nested MIX Stack Assembly – Adding Expansion Modules

Rev. F.1

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.9 Nested MIX Stack Assembly

In Nested configurations, the MIX baseboard and the first MIX expansion module occupy the same VMEbus card cage slot. Expansion module positions are still referred to as "MIX stack position 0" for the first module (the module mounted ***closest to the baseboard***), "stack position 1" for the second module, and "stack position 2" for the third module. While performing these assembly procedures, please refer to Figures C–4, C–7, and C–8, on pages C–8, C–15 and C–17, respectively.

Assemble all MIX components in a static controlled environment!

Improper alignment of the MIX components *WILL* cause failure of the MIX interface and very likely damage the boards!

Disconnect all power to the card cage before inserting or removing any board(s)!

1) Place a MIX baseboard connector on top of the baseboard's MIX contact pattern. Verify that the black stripes on both orange labels are aligned, and that the baseboard connector's keying pin is seated in the keying hole of the baseboard's contact pattern.
2) Place the insulated (shiny) side of a MIX stiffener against the **BOTTOM** of the baseboard's MIX contact pattern.
3) Insert, but ***do not fully tighten***, four flathead screws through the MIX stiffener, through the baseboard, and begin threading them into the baseboard connector.
4) Tighten the flathead screws ***slowly*** and ***evenly***, one turn at a time for each screw in sequence, following the pattern shown in Figure C–7A (at the bottom of this page), to draw the baseboard connector against the baseboard's MIX contact pattern.

   NOTE: Do not torque the flathead screws over 2 in–lbs (Pentek Model #2030 Torque Driver may be used).
5) Place the ***first*** expansion module on the stacking connector, with the component side facing ***DOWN***. ***If this installation requires only one expansion module, proceed to step 14***, on the next page.

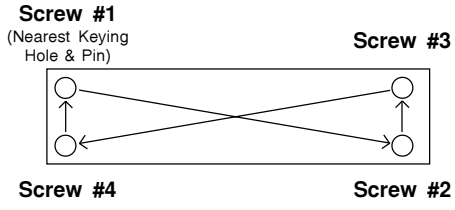


Figure C–7A:
MIX Screw Tightening Pattern

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

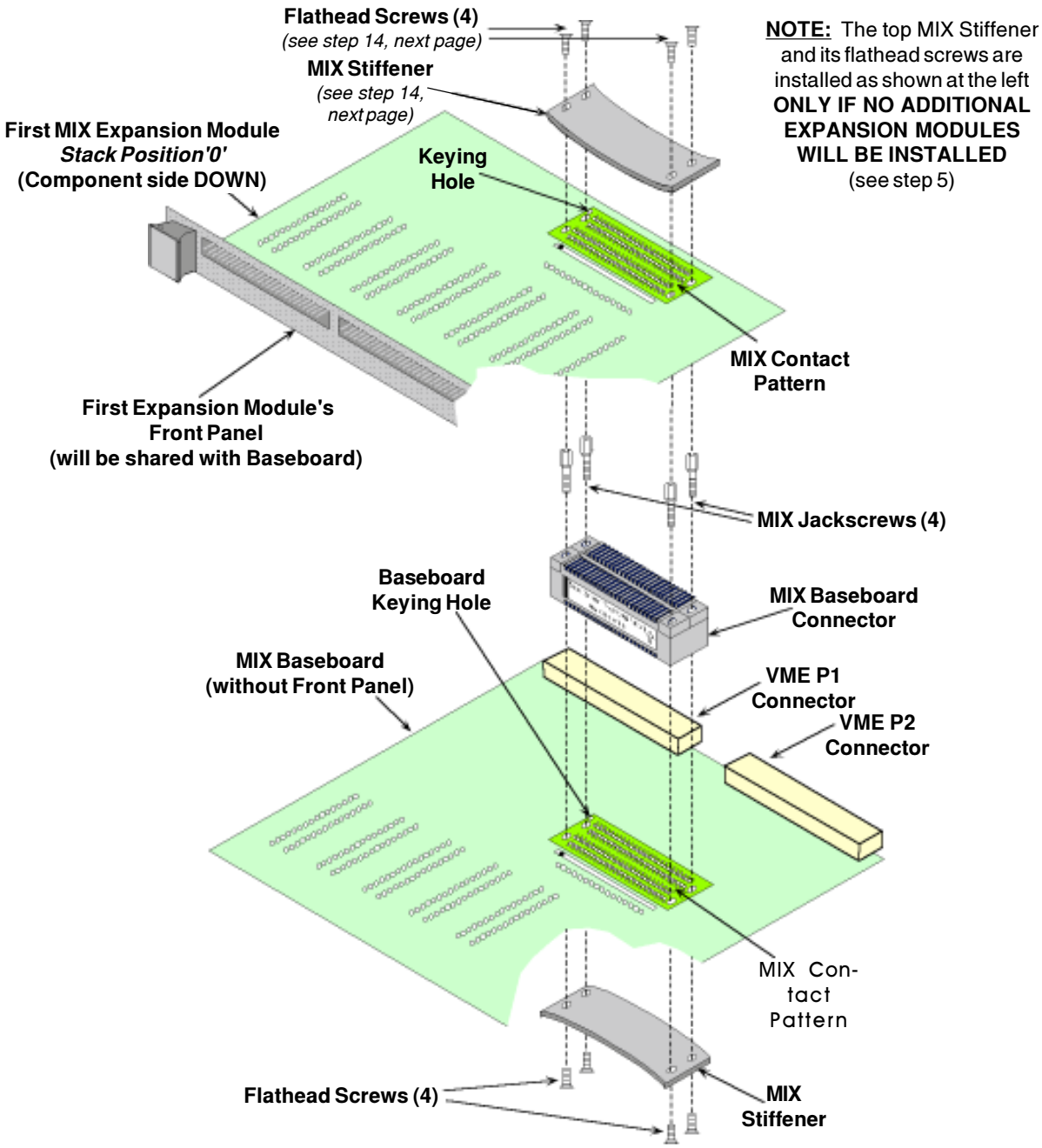


Figure C–7: Nested MIX Stack Assembly – Baseboard and One Expansion Module

Rev. F.1

C.9 Nested MIX Stack Assembly (continued)

6) Place a stacking connector over the MIX contact pattern on the TOP of the ***first*** expansion module (see Figure C–8, on the next page). Verify that the black stripes on all orange labels are aligned, and that the stacking connector's keying pin seats properly in the expansion module's keying hole.

7) Insert, but ***do not fully tighten***, four jackscrews through the stacking connector, through the ***first*** expansion module, and begin threading them into the baseboard connector below the expansion module.

8) Tighten the jackscrews ***slowly*** and ***evenly***, one turn at a time for each screw in sequence, following the pattern shown in Figure C–7A (at the bottom of the page before last), to draw the stacking connector against the expansion module.

   NOTE: Do not torque the jackscrews over 5 in–lbs (Pentek Model #2031 Torque Driver may be used).

9) Place the ***second*** expansion module over the stacking connector with the component side facing DOWN. ***If this installation requires only two expansion modules, proceed to step 14***, below.

10) Place another stacking connector over the MIX contact pattern on the TOP of the ***second*** expansion module (see Figure C–8, on the next page). Verify that the black stripes on all orange labels are aligned, and that the stacking connector's keying pin seats properly in the ***second*** expansion module's keying hole

11) Insert, but ***do not fully tighten***, four jackscrews through the ***second*** stacking connec–tor, through the ***second*** MIX expansion module, and begin threading them into the ***second*** stacking connector.

12) Tighten the jackscrews ***slowly*** and ***evenly***, one turn at a time for each screw in sequence, following the pattern shown in Figure C–7A (at the bottom of the page before last), to draw the stacking connector against the expansion module.

    NOTE: Do not torque the jackscrews over 5 in–lbs (Pentek Model #2031 Torque Driver may be used).

13) Place the ***third*** expansion module over the stacking connector, with the component side facing ***DOWN***.

14) Place the insulated (shiny) side of the MIX stiffener against the MIX contact pattern on the TOP of the ***last*** expansion module.

15) Insert, but ***do not fully tighten***, four flathead screws through the MIX stiffener, through the ***last*** expansion module, and begin threading them into the stacking connector below.

16) Tighten the flathead screws ***slowly*** and ***evenly***, one turn at a time for each screw in sequence, following the pattern shown in Figure C–7A (on the page before last), to draw the stacking connector against the ***last*** expansion module's MIX contact pattern.

    NOTE: Do not torque the flathead screws over 2 in–lbs (Pentek Model #2030 Torque Driver can be used).

This completes the assembly procedure for a nested MIX stack.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

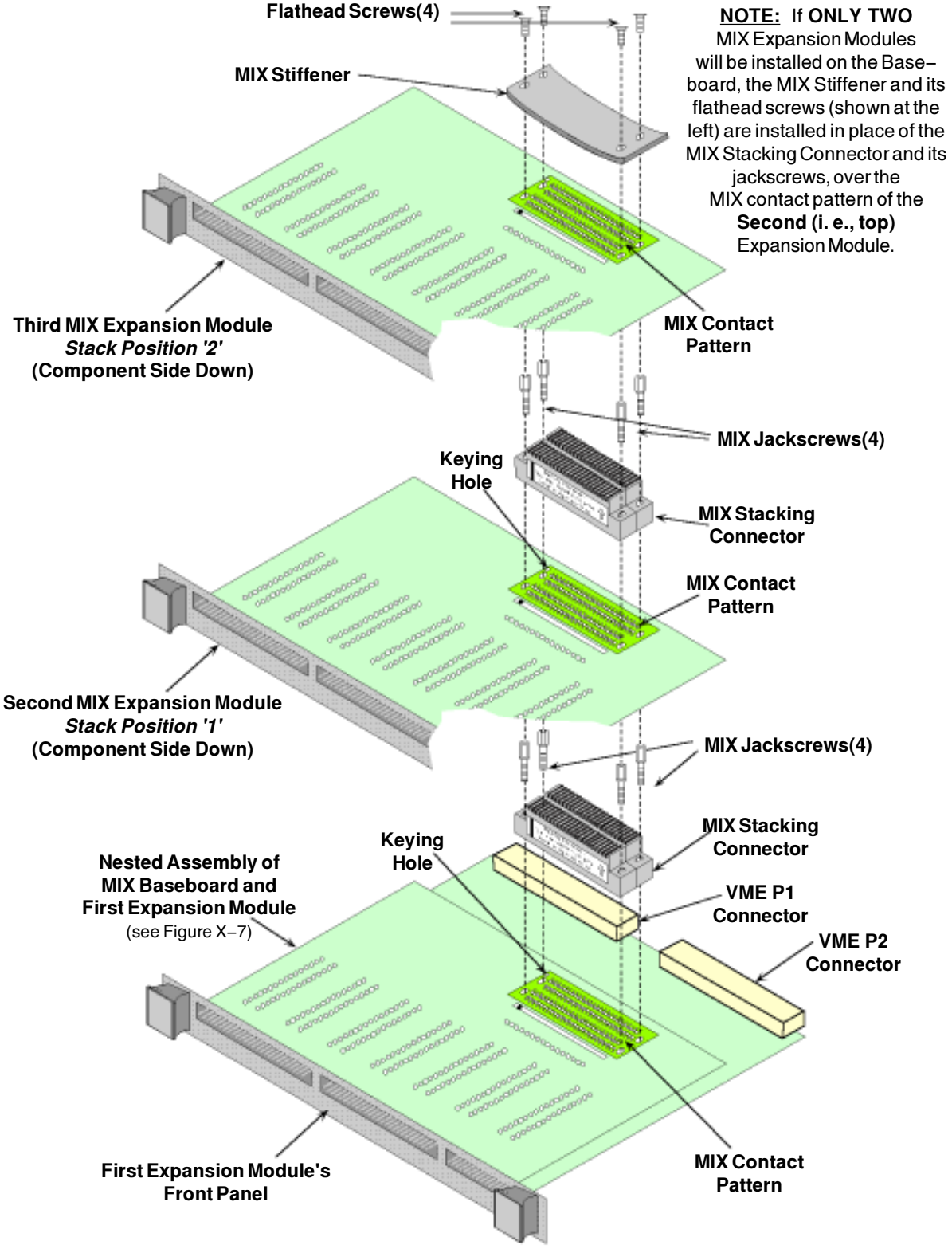


Figure C-8: Nested MIX Stack Assembly - Adding Expansion Modules

Rev. F.1

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.10 MIX Stack Disassembly

When disassembling a MIX stack, loosen all screws in the same manner in which they were tightened (***slowly*** and ***evenly***, one turn at a time for each screw in sequence, following the pattern shown in Figure C–7A, at the bottom of page C–14). **This is of particular importance when loosening the flathead screws**, which are stressed by the spring tension of the MIX stiffener. **Failure to observe this precaution may result in property damage or personal injury, caused by flying screws.**

C.11 Installing Ejector Handles on the MIX Module Front Panel

VME card cage ejector handles should be installed on the front panel of any assembly that engages the VMEbus backplane. For nested installations, where the baseboard's front panel is removed, ejector handles must be installed on the front panel of the nested MIX Module. The installation procedure is described below.

Each Ejector Handle and Panel Mounting Bracket assembly mounts to the front panel with one Philips–head screw and nut per handle. The Philips–head screws thread into hex nuts that are secured in recessed holes within both the top and bottom Ejector Handle and Panel Mounting Brackets. The assembly is illustrated in Figure C–9, on the next page.

1) The Ejector Handles will be mounted on posts on the Panel Mounting Brackets at the top and bottom of the circuit board. These brackets are attached to the front panel with Philips–head screws at the top (and bottom) of the panel, located directly below (and above) the captive card–cage retraining screws. These Philips–head screws are threaded into hex nuts which are recessed within the body of the Panel Mounting Brackets. Locate and remove these two Philips screws. Be sure to retain the screws and nuts, as they will be required to re–attach the Mounting Brackets.
2) Locate the Ejector Mounting Post on the panel bracket (see Figure C–9, Detail B), which is found on the component side of the MIX module. A slotted–head screw is installed in each post. These screws need not be removed.
3) The top and bottom ejector handles differ from one another in the offset of the arm which will pass through the front panel. Holding the handle with the Cover Plate facing you (i. e., with the arm away from you) the top handle will have this arm offset to the left of center when the arm points down. The bottom handle will have the arm offset to the left of center when the arm points up (see Figure C–9A, at the top of the next page). Identify each handle and mount it in its proper position. It may be necessary to gently flex the circuit board to allow the hole in the handle's mounting arm to engage the bracket's mounting post.
4) Slide the Cover Plates out of the Ejector Handles. This will reveal holes in the backs of the handles (again, see Figure C–9A), through which you can replace the top and bottom front panel retaining screws that were removed in step (1).
5) Slide the Cover Plates back into the Ejector Handles (see Figure C–9, Detail A).

This completes the Ejector Handle mounting procedure.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

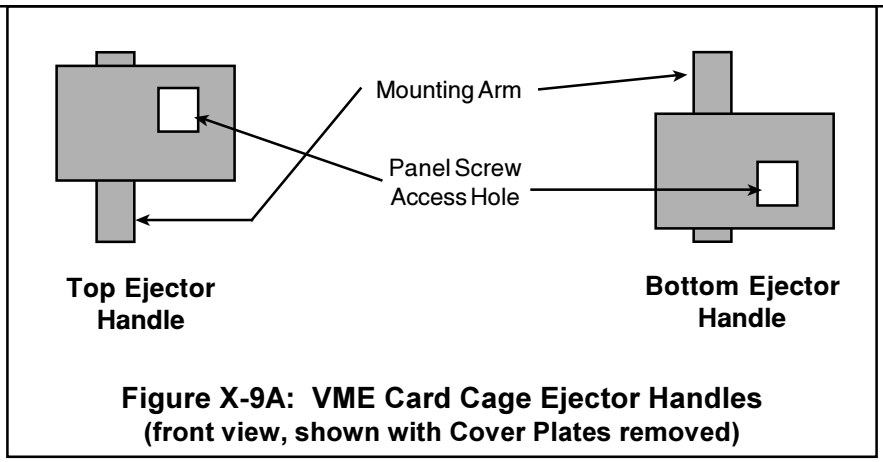


Figure X-9A: VME Card Cage Ejector Handles
(front view, shown with Cover Plates removed)

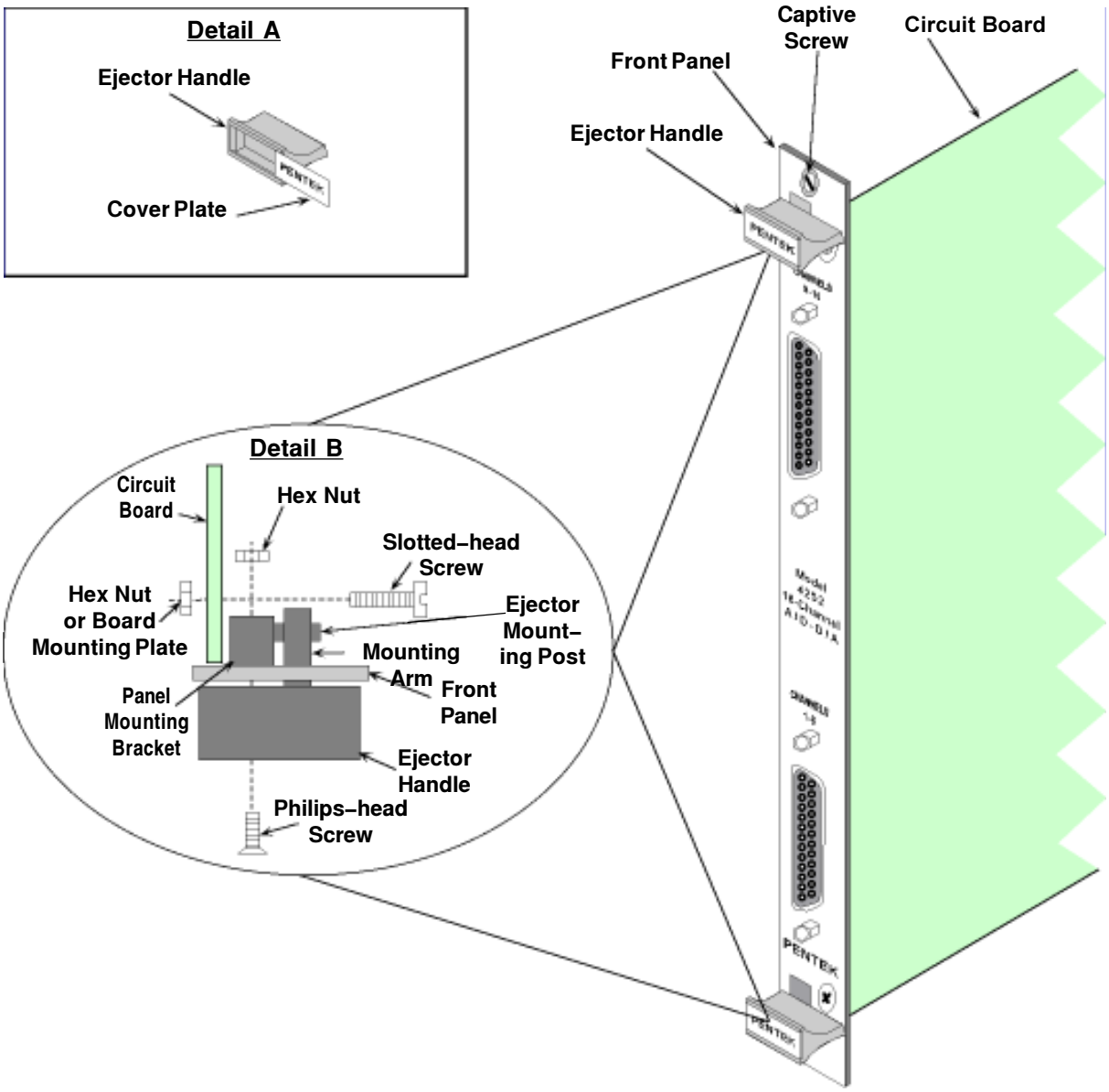


Figure C-9: Installing Ejector Handles on MIX Module Front Panels

Rev. F.1

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.12 Linking MIX Module Front Panels and Ejector Handles

After your MIX stack has been assembled according to the instructions above, the front panels of the modules 'float' until they are secured by their captive screws to the VMEbus card cage. Also, the ejector handle on the Baseboard (or the module nested on the Baseboard) is the only mechanism for removing the MIX stack from the card cage.

In order to provide a more secure mechanical assembly when the stack is removed from the card cage, a pair of 'panel links' is provided with your Baseboard. These links may be used to tie all front panels in the stack together. Use of these links is optional, but strongly recommended if the stack will be removed from the card cage frequently.

Additionally, card cage ejector handles may be installed on all MIX module front panels. These ejectors **MUST** be physically secured to the ejectors on the Baseboard front panel, to assure equal pressure is applied along the card cage ejector plate during the ejection process. Installation of ejectors on the module panels is especially recommended if your stack includes any full–depth modules that engage the VME backplane, such as the Model 4270, the Model 6102, or the Models 4271 or 4272, with Option 012.

For the sturdiest configuration possible, the panel links and module ejector handles can be used together. This assembly method provides for both physical links between all module panels and improved ease of system ejection from the card cage. Procedures for the installation of the panel links and module ejector handle links are provided below. Note that, as a general rule, Pentek does not ship MIX modules with ejector handles installed, but the handles are included with the module. The installation proce–dure for the ejector handles can be found in Section C.11, on page C–18 of this Appen–diC. If you wish to use both panel and ejector handle links, the ejector handles should be installed on the MIX modules **BEFORE** the panel links.

C.12.1 Panel Links

The panel links are machined aluminum bars that may be snapped off to accommodate 2, 3 or 4 front panels in the MIX stack. Two links are required for each stack assembly: one for the top of the panels and one for the bottom. The top and bottom links are symmetrical, but not identical. The silver link braket is used at the top of the panel, and the gold link bracket is used at the bottom. The links are installed behind the front panels, using the screws that hold the plastic Panel Mounting Brackets to the front panel.

Configuring the links for the number of modules in the stack is achieved by snapping the links at the molded indentations, as shown in Figure C–10, on the next page. Grasp the link on either side of the indentation with two pliers and bend the link to snap it to length.

To use the links to join the front panels of the MIX modules, first assemble the MIX stack, using the MIX connectors as described in the assembly in–structions in the preceding sections of this AppendiC. If ejector handles will be used on the modules, install them next, as described in Section C–11 of this AppendiC. Then, follow the steps listed on the next page.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.12 Linking MIX Module Front Panels and Ejector Handles (continued)

C.12.1 Panel Links (continued)

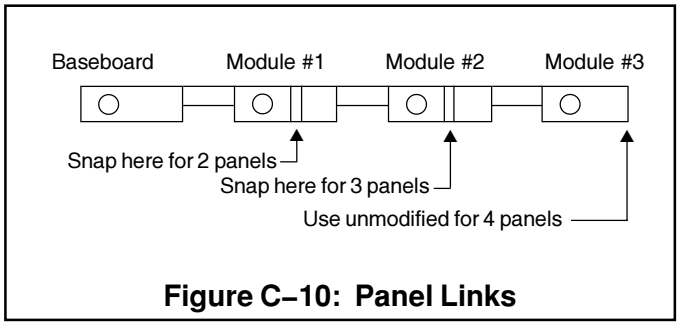


Figure C–10: Panel Links

1) Remove the two cover plates from the ejector handles of the MIX base–board by sliding them sideways. Lay the MIX stack flat on a work sur–face with the baseboard on the bottom.

2) Insert a #0 Philips screwdriver through the hollow center of the baseboard ejectors and loosen the 10 mm long, raised flat–head Philips screws (2.5 mm diam. x 0.45 pitch), located directly behind the two ejectors. Remove the nuts from the plastic brackets behind the front panel. Save these nuts for possible future use. (We suggest that you use the zip lock bag the links came in).

3) Loosen these same screws on each of the MIX modules and remove the nuts from their brackets.

4) Place the links behind the top and bottom plastic brackets such that the holes in the links are aligned with the screw holes on the appropriate panels, as indicated by the caption above the link drawing in Figure C–10, above. Again, use the silver bracket at the tops of the panels, and the gold bracket at the bottoms.

5) Gradually tighten all screws in rotation, a few turns at a time, making sure that the panels are aligned at the top and bottom.

6) If you will **NOT** be using module ejectors, replace the two cover plates in the baseboard ejectors. The MIX stack may now be installed in the card cage, without the module ejectors.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.12 Linking MIX Module Front Panels and Ejector Handles (continued)

C.12.2 Linked Module Ejector Handles

As mentioned in the introductory remarks of this section, ejector handles installed on MIX module front panels must be physically coupled to the ejector handles on the MIX Baseboard. Beginning around October of 1994, all VMEbus ejector handles shipped by Pentek have had small holes drilled in their sides. Ejectors on each unit may be coupled to those on adjacent units by passing rods through these holes. If you have an older MIX Baseboard or module and wish to use linked ejectors, contact the factory at the phone number shown on this manual's title page, and a set of modified ejector handles will be sent to you.

The parts listed below are supplied with each Pentek VMEbus MIX Base–board to support this feature. Figure C–11, at the top of the next page, will help you identify each part.

Quantity	Items	Pentek Part Number
2	ID Plate: VME 2 Slots w/ Logo	305.00031
2	ID Plate: VME 3 Slots w/ Logo	305.00033
2	ID Plate: VME 4 Slots w/ Logo	305.00035
2	Metal Bar Shaft: VME Ejector 2 Slots	384.00020
2	Metal Bar Shaft: VME Ejector 3 Slots	384.00030
2	Metal Bar Shaft: VME Ejector 4 Slots	384.00040
6	Retaining Ring: (E) 0.093 Shaft	388.20000
1	Full Size Bottom Panel Link (Gold)	303.00145
1	Full Size Top Panel Link (Silver)	303.00143

The assembly procedures beginning below assume that you have already installed a pair of handles on each module, that the MIX module stack is fully assembled, and that the ID (or blank aluminum) face plates are NOT inserted in the ejector handles.

1) Select a pair of shafts, based on the number of VME card cage slots your MIX stack will occupy.

2) Align the top and bottom rows of Ejector Handles and insert a shaft through each row. Shafts should be inserted such that the ends with two inscribed slots rest inside the Ejector Handles of the Baseboard (or the module nested on the Baseboard).

c) Install two pairs of retaining 'E' rings on the Shafts inside the Baseboard Ejector Handles. One ring is inserted in each of the two slots on the shafts in the top and bottom Ejector Handles. These rings prevent the shafts from sliding by anchoring the shafts inside the Ejector Handles of the Baseboard (or nested module). Figure C–12, page after next, shows an installed shaft with the retaining rings properly positioned.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.12 Linking MIX Module Front Panels and Ejector Handles (continued)

C.12.2 Linked Module Ejector Handles (continued)

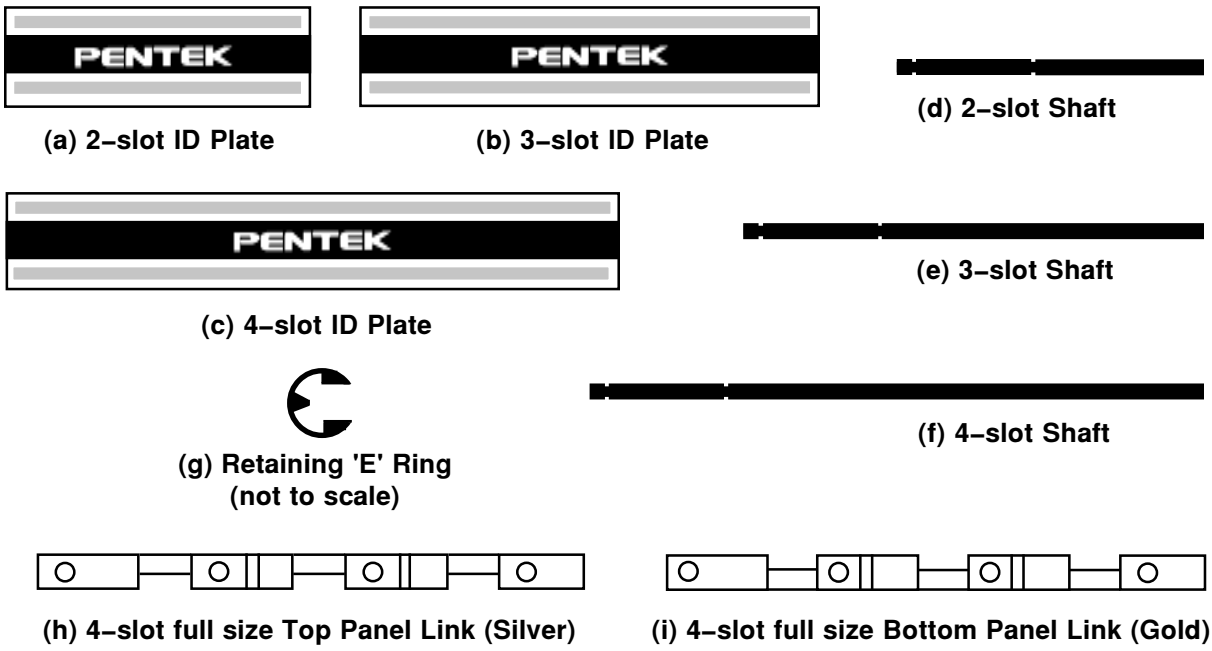


Figure C–11: Linked Module Ejector Parts

d) Select a pair of ID plates (again, based on the number of VME card cage slots your MIX stack will occupy), and install one plate in each row of ejector handles. To install the plates, you may either slide them sideways through all the handles in the row, or position them over the handles and snap them in.

f) The MIX stack may now be installed in the card cage.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

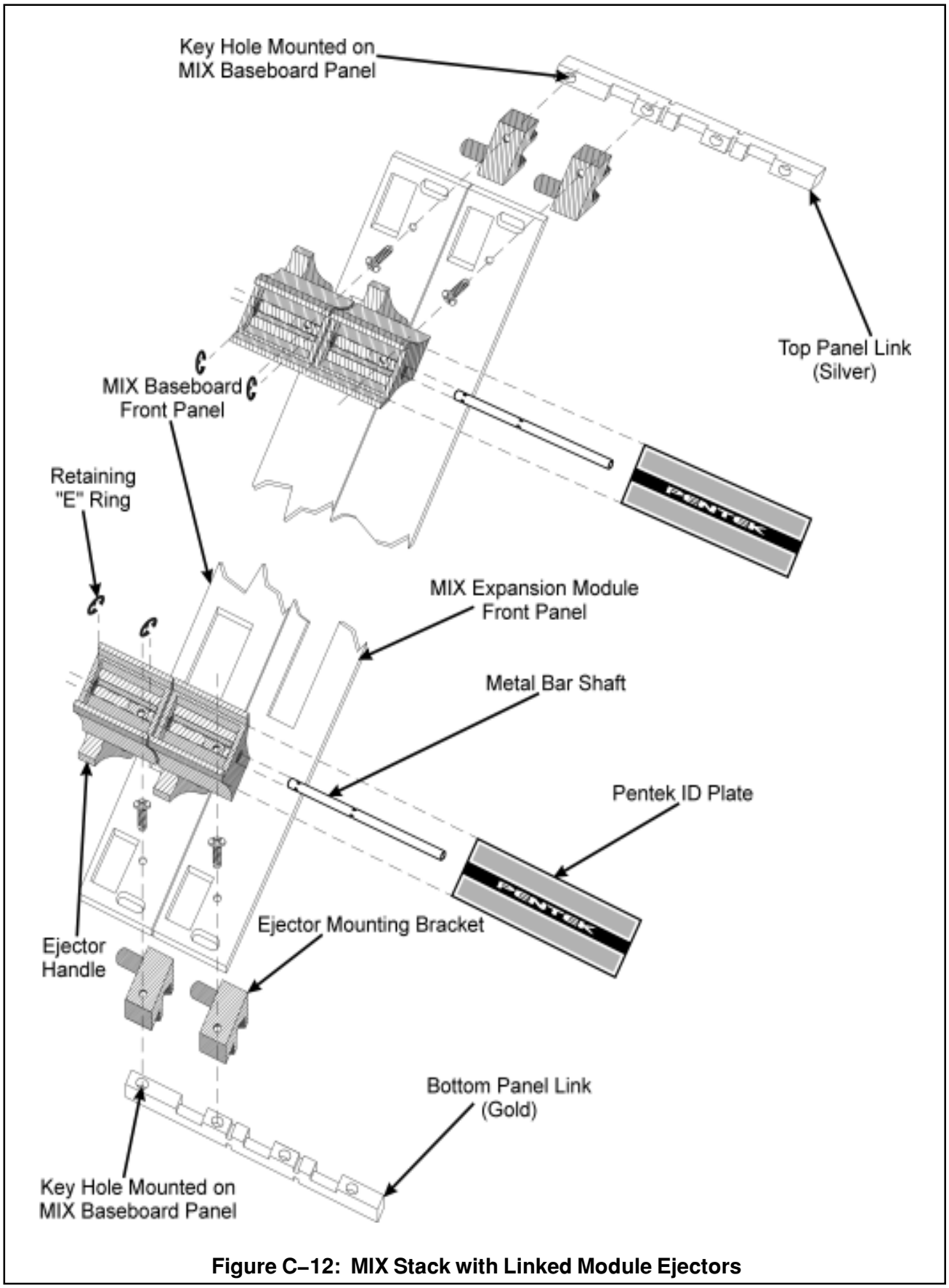


Figure C–12: MIX Stack with Linked Module Ejectors

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

C.13 Installing the Assembly into the VMEbus Card Cage

Once the assembly of the MIX baseboard and expansion module(s) has been completed, the assembly can be installed in the card cage.

If linked module ejector handles have not been installed, and the MIX baseboard will be the only portion of the assembly which actually engages in the back plane, then **use only the front panel ejector/handles of the MIX baseboard** (or nested MIX module) – not those of the expansion modules, when inserting and removing the assembly. After inserting the assembly, push in on the top and bottom ejector/handles of the MIX baseboard or nested module ***only*** to fully seat the connectors in the back plane.

If linked ejectors were installed, and if there is more than one unit in the assembly that engages the backplane, apply equal pressure along the length of the linked ejector handles to seat all units in the VMEbus backplane connectors.

Once the assembly is seated, the captive panels screws at the top and bottom of each front panel should be screwed into the top and bottom rails of the card cage.

When removing the assembly, first loosen **ALL** captive screws on the top and bottom of each front panel. Attempting to remove the MIX stack while any module is still secured to the cage will cause damage to the MIX connectors. Then, push the ejector handles of the assembly away from the center of the panel to eject the assembly from the cage. Once disengaged, pull outward on the ejector handles to remove the assembly.

CAUTION !!
TURN OFF ALL POWER TO THE CARD CAGE
BEFORE INSERTING OR REMOVING ANY BOARD

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

This page is intentionally blank

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Appendix D: Operating Note - A24 Slave Addressing

D.1 Introduction

The VMEbus provides a 32-bit address path. When a VMEbus Master initiates communications with a some Slave devices in A24 space (the 4200/01 among them), the upper 8 address lines (A31 - A24) are not necessarily idle, and the states of these allegedly "unused" bits can be a source of some significant difficulty. The purpose of this note is to prepare you for the potential pitfalls associated with this situation, and to provide some work-arounds for specific problems it presents for the Model 4200/01.

D.2 A24 Address Decoding and the VAC068A

The device which has the responsibility for address decoding in the Model 4200/01 is the VAC068A, manufactured by Cypress Semiconductor. The architecture of the VAC is such that even when the Address Modifier is set for A24 operation, it still reads and decodes the entire 32 bits of VME address information. Thus, when assigning an A24 base address to the Model 4200/01, it is important to know what states will be present on the upper 8 address lines (A31 - A24) during A24 transactions.

These address lines are physically located on the P2 connector on the VME backplane. The signals carried on the P2 connector are considered optional (i. e., they are for "extended" operation) on the VMEbus. Many VME cards, and even some card cages, have no P2 connectors at all. The states of the upper address lines is a function of whether or not the P2 connector is present, and of how the Bus Master deals with them. Three possible cases exist:

1) The 4200/01 is installed in a VME card cage which has no P2 connectors.

2) The 4200/01 is installed in a VME card cage which has P2 connectors, but the Bus Master does not drive the upper 8 address lines.

3) The 4200/01 is installed in a VME card cage which has P2 connectors, and the Bus Master does drive the upper 8 address lines.

D.3 Case 1: No P2 Connectors

In the case of where a 4200/01 installed in a card cage lacking P2 connectors on the VME backplane, the upper 8 address bits are pulled to the low state on the 4200/01 card. In this situation, the first two hex digits of the 32-bit A24 base address assigned to the 4200/01 must be 0x00.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

D.4 Case 2: The Bus Master Does Not Drive the P2 Address Lines

In the case where a Model 4200/01 is installed in a card cage that does have P2 backplane connectors, but the Bus Master does not drive the upper 8 address lines (e. g., if the Master is a Bit3 Model 403 PC/AT - VMEbus adaptor), the A31 - A24 address lines are pulled high by resistors installed on the VME backplane. In this situation, the first two hex digits of the 32-bit A24 base address assigned to the 4200/01 must be 0xFF.

D.5 Case 3: The Upper Address Lines are Driven by the Bus Master

In the third case, the Model 4200/01 is installed in a card cage that has P2 backplane connectors, and the Bus Master does drive the upper 8 address lines (e. g., if the Bus Master is a Heurikon HK68-V3D single-board computer). In such situations, the user must consult the memory map for the master device to determine what will be placed on these address lines.

Taking the above-mentioned Heurikon board as an example, the memory map shows that the on-card address range associated with "Standard" (i. e., A24) VMEbus access is from 0x0100 0000 through 0x01FF FFFF. Typically, the local address prefix flows out onto the VMEbus address lines during Master transactions. Thus, if this device was to master VME transactions with the Model 4200/01, the first two hex digits of the 32-bit A24 base address assigned to the 4200/01 must be 0x01.

As an additional example, if one 4200/01 masters an A24 transaction with another 4200/01, the address prefix would need to be 0xF4 (refer to the 68030 address map in Section 3.2.2 of this manual).

D.6 Flash EEPROM Address Conflict

If the requirements of your application are such that the A24 base address you assign to the Model 4200/01 begins with the prefix 0xFF in the two most significant hex digits of the 32-bit address, a conflict will exist between the memory regions assigned to A24 SRAM access and to the Flash EEPROMs (which reside in the region between 0xFF00 0000 and 0xFF07 FFFF). This will result in an apparent inability of the 4200/01's 68030 to access the Flash Memory.

This situation can be worked around by disabling Slave Access when attempting to access EEPROM. To accomplish this, write a '0' to bit 3 of the VAC's decode control register, at 68030 address 0xFFFD 1400. Remember to set this bit back to the '1' state before attempting subsequent SRAM access.

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

Appendix E: The 4200/01 Distribution Diskette

E.1 Introduction

The diskette that was enclosed with this manual contains all the example programs that were referred to in the body of this manual, and several others as well. "C" language source code is provided that demonstrates how one would work with Pentek's most popular MIX module products when thay are mounted on a Model 4200 or 4201 MIX Baseboard.

The root directory of the diskette contains an archived file, **4200.zip**, and two un-archiving utilities (**pkunzip.exe** and **unzip**). A **readme** file is also included, that tells which unarchiver to use and gives the appropriate command syntax, depending on the type of computer you are using. The **readme** file is listed below.

```
To restore directory set on a PC running DOS

      pkunzip -d 4200.zip

To restore directory set on a SUN running UNIX

      unzip -a -x 4200.zip
```

E.2 Contents of 4200.zip

When the unarchiving process has been completed, the directory in which **4200.zip** was unarchived will contain nine new directories and a new text file, **readme.1st**, in addition to the original archive. Below is a list of the directory names and a description of their contents.

Directory	Description
\4248	Sample application to demonstrate data collection from a Pentek Model 4248 32 Channel A/D Converter on a Model 4200/01 MIX baseboard
\4252	Sample application to demonstrate data collection and output from a Pentek Model 4252 16 Channel A/D-D/AConverter on a Model 4200/01 MIX baseboard
\4253	Sample application to demonstrate data output from a Pentek Model 4253 32 Channel D/A Converter on a Model 4200/01 MIX baseboard
\blink	Sample application to blink LED on Model 4200/01 front panel
\demo	Sample application to demonstrate Model 4200/01 Standard I/O capabilities

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

E.2 Contents of 4200.zip (continued)

Directory	Description
\dma	Sample application to demonstrate DMA capabilities of Model 4200/01
\include	Assorted header files used by the other sample applications
\rel2.0	Contains the compiled code in the 4200/01's Boot Flash EEPROM, and a compiled program to program the flash EEPROMs, in Motorola S record format. Also contains a subdirectory, **\src**
\rel2.0\src	Source code, etc. for the compiled files in the parent directory
\util	Contains another **readme.1st** text file and four subdirectories, **\pc-bit3**, **\sbus-bt3**, **\vme** and **\vxworks**, each of which contain a loader for the 4200/01 and the host monitor program discussed in Section 2.4.14 of this manual. These will be discussed further in Section B.3.

The content of the **readme.1st** file in the main directory is listed below.

```
Model 4200/01 Demo Disk

This disk contains the necessary tools and examples to load and
run programs on the Model 4200/01 Mix Module Baseboard.  All
examples have been compiled on a Sun workstation running Sun OS
ver 4.1.x using the GNU 'C' compiler toolkit provided by :

               Free Software Foundation, Inc.
               675 Mass Ave,
               Cambridge, MA 02139 USA

Getting Started
---------------

1) Connect an RS-232 Terminal or Serial Port on any PC or work-
   station capable of running terminal emulation software to se-
   rial port A or Monitor Port.  The Serial Port default communi-
   cation parameters should be set to 9600 Baud, No Parity, 8
   Data Bits, and 1 Stop Bit.  Refer to section 2.4 in the 4200/
   01 manual for a description of features available over the
   monitor port.

2) Select the desired base address settings.  Refer to section
   2.2.2 in the 4200/01 manual to select Base Address settings.

3) Press Reset on the 4200/01's Front Panel and verify the base
   address settings are displayed along with the PENTEK logo on
   the RS-232 Terminal.
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

E.2 Contents of 4200.zip (continued)

```
4) Run the Blink and Demo Programs using one of the loaders pro-
   vided in the UTIL directory.


Upgrading 4200/01 Firmware
--------------------------

This disk contains the executable and source code for Revision
2.0 Firmware for the 4200/01.  Firmware can be upgraded two ways:

        Serial Port(Revision 1.84 or Later)
        VME

Note:   Be aware that downloading new firmware to the 4200/01
        will erase the unit's existing boot code.  User-generated
        code should NOT be programmed in this manner.  If the
        4200/01 doesn`t boot after performing the upgrade (LED
        stays off upon RESET), contact PENTEK at (201)767-7100.

Upgrading Firmware over VME
---------------------------

1) Before you try to load the new firmware, verify that programs
   can be loaded using one of the loaders provided on this disk.
   They may be different from the one you currently use.

2) Enable the Flash EEPROM to be written by moving the jumper on
   the JB5 jumper block to position 1-2.

3) Loading the firmware consists of 2 steps:

   Loading the Bootcode to SRAM at location Base Address +
   0x10000

   Loading and Executing a program which erases the Flash EEPROM
   and writes the new Bootcode. This program is loaded in SRAM at
   location Base Address + 0x50008.

4) Enter the following commands:
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

E.2 Contents of 4200.zip (continued)

```
load42 -l -pxxx1 -myy bootrom.hex

Where:
   xxx1 -  Bits A31-A16 of the base address for the SRAM.
           Bits A31-A24 only need to be specified for A32 ad-
           dress mode.

   yy   -  Addressing Mode (24 or 32)

The loader may request that a key be pressed after the 4200/01
has completed Resetting.

load42 -x -pxxxx -myy progrom.hex

This program will be executed after it is loaded.  On the Se-
rial Port Terminal you will see the following messages:

      Erasing Device...Done
      Verifying Erase...Done
      Now Programming rom...Done
      Verifying...Done

The 4200/01 will reset and re-display the logo along with the
current revision number.

Example:
      To load bootcode to a 4200/01 with it's Base Address at
      0x80000000 in A32 Addressing Mode.
         load42 -l -p8001 -m32 bootrom.hex
         load42 -x -p8000 -m32 progrom.hex

      To load bootcode to a 4200/01 with it's Base Address at
      0x800000 in A24 Addressing Mode.
         load42 -l -p81 -m24 bootrom.hex
         load42 -x -p80 -m24 progrom.hex

5) Replace the Write Protect jumper on JB5 to positions 2-3.


Upgrading Firmware over Serial Port
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

E.2 Contents of 4200.zip (continued)

```
-----------------------------------

1) Enable the Flash EEPROM to be written by moving the jumper on
   the JB5 jumper block to position 1-2.

2) Select the Upgrade Firmware Option in the 4200/01 monitor's
   menu.

3) Upload the program selecting ASCII Protocols and the file to
   be uploaded.

Note:   The following parameters for ASCII transfers need to be
        specified in the terminal emulation software being used:

        Carriage Return translation - None
        Line Feed translation           - None
        Space between sending each line - .2 seconds
        ECHO locally                    - No

4) Upon successful transfer, the Monitor will provide feedback on
   the number of bytes loaded and request whether to Upgrade the
   Firmware or Abort.  If you select the Program option, the fol-
   lowing will display on the terminal.

        Erasing Device...Done
        Verifying Erase...Done
        Now Programming rom...Done
        Verifying...Done

   The 4200/01 will reset and re-display the logo along with the
   current revision number.

5) Replace the Write Protect jumper on JB5 to positions 2-3.
```

E.3 Utility Programs for the Model 4200/01

The **\util** directory that was created when **4200.zip** was unarchived contains subdirectories whose names describe the various host system connections currently in use among most of Pentek's customer base. Those directories each contain source and executable files for a program to download code to the Model 4200/01 (**load42.c** and **load42.exe**), and for the host monitor program descrbed in Section 2.4.14 of this manual (**mon42.c** and **mon42.exe**). There is also another **readme.1st** text file, to describe the use of these programs. A listing of that files contents begins at the top of the next page.

```
        4200/01 Utility Programs
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

E.3 Utility Programs for the Model 4200/01 (continued)

```
This disk contains a Loader and VME Monitor program for the fol-
lowing Platforms:

        Embedded Sparc or Sun running SUN OS
        Embedded Processor running VxWorks
        Sun with a Bit 3 SBUS to VME Adapter
        PC with a Bit 3 PC to VME Adapter

Load42 Utility
--------------

This program enables the user to download and execute programs in
the 4200/01's SRAM.  Programs must be in the Motorola 'S' Record
Format.

To Invoke:

   load42 -xlmp file

   Arguments:

   -x      Do Not Perform a VME Reset prior to loading program.
           If a reset is performed, a request will be made to
           press any key to start.  This will allow the 4200 to
           finish initializing itself.

   -l      Load the program only.  Do not execute

   -m      Specify addressing mode(24 or 32).

   -p      Specify the upper 8 address bits.  This can also be
           used to offset the program in SRAM.

   file    Executable in Motorola 'S' record format.

To Compile:

For   Sun OS:    cc load42.c -o load42
      PC:        Used Turbo 'C', may need to be modified for other
                 compilers
      VxWorks:   Used GNU Compiler


Monitor Utility
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

E.3 Utility Programs for the Model 4200/01 (continued)

```
---------------

This program enables the user to command the 4200/01's built in
monitor to perform certain functions.  The available functions
are listed in  Section 2.4.14 in the 4200/01 Manual.


For   Sun OS:    cc mon42.c -o mon42
      PC:        Used Turbo 'C', may need to be modified for other
                 compilers
      VxWorks:   Used GNU Compiler
```

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

This page is intentionally blank

Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

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