

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 *Concurrent Technologies VP PSE/C13* at our website: ***Click HERE***

Technical Reference Manual for VP PSE/C1x Celeron® and VP PSE/P3x Pentium® III VME Bus Single Board Computer

Manual Order Code 550 0009-10 Rev 07 August 2001

Concurrent Technologies Inc
3840 Packard Road
Suite 130
Ann Arbor, MI 48108
USA
Tel: (734) 971 6309
Fax: (734) 971 6350

Concurrent Technologies Plc
4 Gilberd Court
Newcomen Way
Colchester, Essex CO4 4WN
United Kingdom
Tel: (+44) 1206 752626
Fax: (+44) 1206 751116

E-mail: info@gocct.com http://www.gocct.com

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

NOTES

Information furnished by Concurrent Technologies is believed to be accurate and reliable. However, Concurrent Technologies assumes no responsibility for any errors contained in this document and makes no commitment to update or to keep current the information contained in this document. Concurrent Technologies reserves the right to change specifications at any time without notice.

Concurrent Technologies assumes no responsibility either for the use of this document or for any infringements of the patent or other rights of third parties which may result from its use. In particular, no license is either granted or implied under any patent or patent rights belonging to Concurrent Technologies.

Some parts of this document are reproduced with the permission of and remain copyright Phoenix Technologies Ltd, 1997.

No part of this document may be copied or reproduced in any form or by any means without the prior written consent of Concurrent Technologies.

All companies and product names are trademarks of their respective companies.

CONVENTIONS

Throughout this manual the following conventions will apply:
** or* $\overline{}$ *over a name represents an active low signal. e.g. INIT* or* $\overline{INIT}$
h denotes a hexadecimal number. e.g. FF45h
byte represents 8-bits
word represents 16-bits
dword represents 32-bits

GLOSSARY OF TERMS

BIOS · · · · · · · · *Basic Input Output System*
BIST · · · · · · · · *Built In Self Test*
BSB · · · · · · · · *Back Side Bus*
CCT · · · · · · · · *Concurrent Technologies*
CPU · · · · · · · · *Central Processing Unit*
DFP · · · · · · · · *Digital Flat Panel*
DMA · · · · · · · · *Direct Memory Access*
ECP · · · · · · · · *Extended Capabilities Port*
EIDE · · · · · · · · *Enhanced Integrated Drive Electronics*
EPP · · · · · · · · *Enhanced Parallel Port*
EPROM · · · · · · *Electrically Programmable Read Only Memory*
FSB · · · · · · · · *Front Side Bus*
ISA · · · · · · · · *Industry Standard Architecture*
PCI · · · · · · · · *Peripheral Component Interconnect*
PIIX4E · · · · · · · *PCI ISA IDE Xcelerator*
PMC · · · · · · · · *PCI Mezzanine Card*
POST · · · · · · · · *Power-on Self Test*
RFU · · · · · · · · *Reserved for Future Use*
SCC · · · · · · · · *Serial Communications Controller*
SCSI · · · · · · · · *Small Computer System Interface*
SDRAM · · · · · · *Synchronous Dynamic Random Access Memory*
SGDRAM · · · · · *Synchronous Graphics DRAM*
SODIMM · · · · · *Small Outline Dual Inline Memory Module*
UART · · · · · · · *Universal Asynchronous Receiver Transmitter*
USB · · · · · · · · *Universal Serial Bus*
VME · · · · · · · · *Versa Module Europe*

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

NOTATIONAL CONVENTIONS

NOTE	Notes provide general additional information.
WARNING	Warnings provide indication of board malfunction if they are not observed.
CAUTION	Cautions provide indications of board or system damage if they are not observed.

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

Revision	Revision History	Date
01	First Release	January 2000
02	Update for Rev C Board and 600MHz Pentium III	May 2000
03	Update for 700 and 850MHz Pentium III	July 2000
04	Corrections to Figure 3-6 and F-1	July 2000
05	Amendments to Chapter 8	August 2000
06	Included DiskOnChip and Remote Boot BIOS information	March 2001
07	Updated for: Rev D board, support for 384MBytes SDRAM, DiskOnChip and Remote Boot BIOS information, new BIOS VME setup screens, and new P2 breakouts; merged VP PSE/C1x and VP PSE/P3x manuals into this one	August 2001

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

Table of Contents

1. Introduction and Overview 1-1

1.1 General 1-1
1.2 The VP PSE/P3x - Main Features 1-2
1.2.1 Central Processor 1-2
1.2.2 Memory Controller 1-2
1.2.3 Level 2 Cache 1-2
1.2.4 SDRAM 1-2
1.2.5 PCI Bus and Bridges 1-2
1.2.6 EPROM 1-2
1.2.7 Serial Communications 1-2
1.2.8 SCSI Controller 1-2
1.2.9 EIDE Controller 1-3
1.2.10 Ethernet Controller 1-3
1.2.11 Graphics Controller 1-3
1.2.12 VME Interface 1-3
1.2.13 Keyboard & Mouse 1-3
1.2.14 Real Time Clock 1-3
1.2.15 Floppy Disk 1-3
1.2.16 Parallel Interface 1-3
1.2.17 USB 1-3
1.2.18 DiskOnChip® Site 1-3
1.2.19 PMC Interface 1-3
1.3 Additional Board Options 1-4

2. Installation 2-1

2.1 General 2-1
2.2 Unpacking and Inspection 2-1
2.3 Working Environment 2-1
2.4 Installation and Power-up 2-2

3. Configuration 3-1

3.1 General 3-1
3.2 Factory Setting 3-1
3.3 Battery Installation/Replacement 3-3
3.4 SCSI Interface Functions 3-4
3.5 Console Mode 3-5
3.6 VSA Mode Jumper 3-6
3.7 Test Jumpers 3-7
3.8 Watchdog Jumper 3-8
3.9 Front Panel Switch Jumper 3-9
3.10 VME System Reset Jumper 3-10

4. Programming Examples 4-1

4.1 PCI Bus Interface 4-1
4.2 Enabling the Control/Status Registers 4-7
4.3 Programming the Watchdog 4-8
4.4 Intel (Digital) 21143 Ethernet Controller 4-10
4.4.1 Ethernet Data Definitions 4-27
4.5 SCSI Processor Interface Routines 4-32
4.6 VME to PCI Configuration 4-42
4.7 Warm Reset via PORT-92 or Keyboard Controller 4-51

5. Functional Description 5-1

5.1 General 5-1
5.2 Functional Description 5-1

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

5.3 Memory Structure · · · · · 5-2
5.3.1 Memory Maps · · · · · 5-3
5.3.2 Memory Controller · · · · · 5-3
5.3.3 SDRAM · · · · · 5-3
5.3.4 PC BIOS Flash EPROM · · · · · 5-3
5.3.5 Off-board VME Memory · · · · · 5-4
5.3.6 VME Byte Swapping · · · · · 5-4
5.4 I/O Functions · · · · · 5-5
5.4.1 PCI ISA IDE Xcelerater (PIIX4E) · · · · · 5-5
5.4.1.1 Interrupt Controllers · · · · · 5-6
5.4.1.2 PCI Interrupts · · · · · 5-7
5.4.1.3 NMI · · · · · 5-7
5.4.1.4 Timers · · · · · 5-7
5.4.1.5 EIDE Interface · · · · · 5-7
5.4.2 Super I/O Controller · · · · · 5-8
5.4.3 Real Time Clock (RTC) · · · · · 5-8
5.4.4 Keyboard & Mouse Controllers · · · · · 5-8
5.4.5 Ethernet Controller · · · · · 5-8
5.4.6 Graphics Interface · · · · · 5-8
5.4.7 Flat Panel Interface · · · · · 5-8
5.4.8 DiskOnChip · · · · · 5-8
5.4.9 SCSI Controller · · · · · 5-9
5.5 443BX Chipset Registers · · · · · 5-10
5.5.1 Configuration Space Enable Register/Configuration Address Register · · · · · 5-10
5.5.2 Configuration Data Register · · · · · 5-10
5.5.3 PCI Arbiter · · · · · 5-10
5.6 Front Panel LEDs · · · · · 5-11
5.6.1 Run LED (R) Green · · · · · 5-11
5.6.2 POST LED (P) Yellow · · · · · 5-11
5.6.3 100 Mbit Enabled (100) Yellow · · · · · 5-11
5.6.4 Link/Activity LED (LK/ACT) Green · · · · · 5-11
5.6.5 Switch (SW) · · · · · 5-11

6. PC BIOS Firmware · · · · · 6-1

6.1 General · · · · · 6-1
6.2 Setup Options · · · · · 6-1
6.2.1 Menu Bar · · · · · 6-1
6.2.2 Legend Bar · · · · · 6-2
6.2.3 Field Help Window · · · · · 6-2
6.2.4 General Help Window · · · · · 6-3
6.2.5 Main Setup Menu · · · · · 6-4
6.2.5.1 Main: Primary Master and Primary Slave Sub-Menus · · · · · 6-6
6.2.5.2 Main: Memory Cache Sub-Menu · · · · · 6-8
6.2.6 Advanced Setup Menu · · · · · 6-10
6.2.6.1 Advanced: Advanced Chipset Control · · · · · 6-11
6.2.6.2 Advanced: I/O Device Configuration · · · · · 6-12
6.2.7 Security Setup Menu · · · · · 6-13
6.2.8 Universe Setup Menu · · · · · 6-14
6.2.8.1 Universe: PCI Slave Images · · · · · 6-16
6.2.8.2 Universe: VME Slave Images · · · · · 6-17
6.2.9 Boot Setup Menu · · · · · 6-18
6.2.9.1 Boot: Boot Device Priority · · · · · 6-19
6.2.9.2 Boot: Hard Drive · · · · · 6-20
6.2.10 Exit Menu · · · · · 6-21
6.3 POST Errors · · · · · 6-22
6.4 Serial Console · · · · · 6-23
6.5 SCSI BIOS · · · · · 6-24
6.5.1 Boot Initialization with BIOS Boot Specification (BBS) · · · · · 6-24

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

6.5.2 CDROM Boot Initialization 6-24
6.6 Starting the SCSI BIOS Configuration Utility 6-25
6.6.1 Main Menu 6-26
6.6.2 Change Adapter Status 6-26
6.6.3 Adapter Boot Order 6-27
6.6.4 Additional Adapter Configuration 6-27
6.6.5 Display Mode 6-27
6.6.6 Mono/Color 6-27
6.6.7 Language 6-27
6.6.8 Help 6-27
6.6.9 Quit 6-27
6.6.10 Esc 6-28
6.6.11 Utilities Menu 6-28
6.6.11.1 Adapter Setup Menu 6-28
6.6.11.2 Device Selections Menu 6-31
6.6.12 To Exit the SCSI BIOS Configuration Utility 6-33
6.7 Network Boot ROM 6-34

7. VME System Architecture Test Handler 7-1

7.1 Introduction 7-1
7.2 The VSA Environment 7-1
7.2.1 Slot Numbering 7-1
7.2.2 VSA Console Devices 7-1
7.2.3 Starting the Master Test Handler 7-1
7.2.4 Remote Testing from the System Controller 7-2
7.2.5 Bootloading the BIOS 7-2
7.2.6 BIST Execution 7-2
7.3 MTH Command Reference 7-3
7.3.1 Help Screens 7-3
7.3.2 General Commands 7-3
7.3.3 Utility Commands 7-5

8. VSA Mode Diagnostics 8-1

8.1 Initialization Checks 8-1
8.1.1 Check 16: CPU Alive Check 8-1
8.1.2 Check 18: Scratchpad RAM Check 8-1
8.2 BIST Descriptions 8-2
8.2.1 Test 1: Test Initialization Routine 8-2
8.2.2 Test 2: PROM Check 8-2
8.2.3 Test 4: Numeric Coprocessor Test 8-2
8.2.4 Test 6: Interconnect Image Check 8-3
8.2.5 Test 7: Off-board Interconnect Access 8-3
8.2.6 Test 9: 8254 PIT Test 8-3
8.2.7 Test 10: 8259A PIC Test 8-3
8.2.8 Test 12: Local RAM Fixed Pattern Test 8-3
8.2.9 Test 13: SCC Access 8-4
8.2.10 Test 20: Universe NMI Test 8-4
8.2.11 Test 23: Local RAM Read/Write Test 8-4
8.2.12 Test 25: Local RAM Dual Address Test 8-4
8.2.13 Test 27: Local RAM Execution Test 8-5
8.2.14 Test 28: SCC Interrupt Test 8-5
8.2.15 Test 29: SCC Internal Loopback Test 8-5
8.2.16 Test 30: SCC External Loopback Test 8-6
8.2.17 Test 32: 690x0 SVGA Controller Test 8-6
8.2.18 Test 33: Universe PCI–>VME Test 8-6
8.2.19 Test 34: Universe PCI Config Utility 8-7
8.2.20 Test 35: Universe VME Config Utility 8-7
8.2.21 Test 36: VME Bus Byte Swapping 8-7

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

8.2.22 Test 38: SCSI Processor Self-Test · · · · · 8-8
8.2.23 Test 39: SCSI Interface · · · · · 8-8
8.2.24 Test 48: SCSI Hard Disk Drive Ready · · · · · 8-8
8.2.25 Test 49: SCSI Floppy Disk Drive Ready · · · · · 8-8
8.2.26 Test 50: SCSI Tape Drive Ready · · · · · 8-9
8.2.27 Test 51: SCSI Hard Disk Drive Read Test · · · · · 8-9
8.2.28 Test 52: SCSI Floppy Disk Read Test · · · · · 8-9
8.2.29 Test 53: SCSI Tape Drive Read Test · · · · · 8-10
8.2.30 Test 54: SCSI Inquiry · · · · · 8-10
8.2.31 Test 55: SCSI Reset Test · · · · · 8-10
8.2.32 Test 56: IDE Controller Test · · · · · 8-11
8.2.32.1 Register Access Test · · · · · 8-11
8.2.32.2 Controller Diagnostics Test · · · · · 8-11
8.2.32.3 Identify Disk Drive · · · · · 8-11
8.2.33 Test 57: SCSI Fixture Test · · · · · 8-12
8.2.34 Test 58: IDE Fixture Test · · · · · 8-12
8.2.35 Test 64: PC Keyboard Test · · · · · 8-13
8.2.36 Test 67: Printer Port Test · · · · · 8-13
8.2.37 Test 68: Real Time Clock Test · · · · · 8-14
8.2.38 Test 69: Digital 21143 100MHz LAN · · · · · 8-15
8.2.39 Test 70: Maxim 1617 Thermal Sensor Test · · · · · 8-16
8.2.39.1 Basic Functionality · · · · · 8-16
8.2.39.2 Temperature Readout · · · · · 8-16
8.2.39.3 Set Alarms · · · · · 8-16
8.2.39.4 Change Update Frequency · · · · · 8-17
8.2.39.5 Full Readout · · · · · 8-18
8.2.40 Test 71: DEC 21143 Interface Test · · · · · 8-19
8.2.41 Test 80: SCSI Based PMC Site Test · · · · · 8-19
8.2.42 Test 85: Floppy Disk Drive Test · · · · · 8-19
8.2.42.1 Controller Access Test · · · · · 8-19
8.2.42.2 Diskette Access Test · · · · · 8-19
8.2.42.3 Disk Checksum Test · · · · · 8-19
8.2.43 Test 101: Display Memory Utility · · · · · 8-20
8.2.44 Test 102: Fill Memory Utility · · · · · 8-20
8.2.45 Test 103: I/O Read Utility · · · · · 8-20
8.2.46 Test 104: I/O Write Utility · · · · · 8-20
8.2.47 Test 105: Interconnect Read Utility · · · · · 8-21
8.2.48 Test 106: Interconnect Write Utility · · · · · 8-21
8.2.49 Test 107: Cache Control Utility · · · · · 8-21
8.2.50 Test 120: PCI Configuration Utility · · · · · 8-21
8.2.51 Test 121: PCI Read Utility. · · · · · 8-22
8.2.52 Test 122: PCI Write Utility. · · · · · 8-22
8.2.53 Test 126: Display Board Configuration · · · · · 8-22
8.2.54 Test 127: Retrieve BIST Information · · · · · 8-22

A. Appendix A · · · · · A-1

A.1 Functional Specification · · · · · A-1
A.2 Connectors · · · · · A-2
A.2.1 Serial Interface Pin-Outs · · · · · A-3
A.2.2 VME Interface (P1) Pin-outs · · · · · A-4
A.2.3 Auxiliary Connector (P2) Pin-outs (Wide SCSI, Panel Link & EIDE) · · · · · A-5
A.2.4 Auxiliary Connector (P2) Pin-outs (Narrow SCSI, EIDE & Printer) · · · · · A-6
A.2.5 Auxiliary Connector (P2) Pin-outs (EIDE Only) · · · · · A-7
A.2.6 Keyboard Connector (P5) Pin-outs · · · · · A-8
A.2.7 Mouse Connector (P4) Pin-outs · · · · · A-9
A.2.8 PMC Expansion Carrier Connector · · · · · A-10
A.2.9 PMC Connector · · · · · A-12
A.2.10 Ethernet Interface (P11) Pin-outs · · · · · A-14

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

A.2.11 Processor Debug Port (P10) Pin-outs . . . A-15
A.2.12 Universal Serial Bus (P6) Pin-outs . . . A-16
A.2.13 VGA (J3) Pin-outs . . . A-17
A.3 Environmental Specification . . . A-18
A.3.1 Temperature Range . . . A-18
A.3.2 Humidity . . . A-18
A.4 Dimensions . . . A-18
A.5 Electrical Specification . . . A-18
A.5.1 Power Supply Requirements . . . A-18

B. Quick Reference Guide . . . B-1

B.1 Memory Addressing Summary . . . B-1
B.1.1 Real Mode Map . . . B-1
B.1.2 Protected Mode . . . B-1
B.1.3 I/O Addressing Summary . . . B-2
B.2 I/O Functions . . . B-3
B.2.1 Interrupt Controllers . . . B-3
B.2.2 PCI Interrupts . . . B-4
B.2.3 NMI . . . B-4
B.2.4 Onboard Status & Control Registers . . . B-5
B.2.4.1 Status & Control Register 0 . . . B-5
B.2.4.2 Status & Control Register 1 . . . B-6
B.2.4.3 Watchdog Status & Control Register . . . B-7
B.2.4.4 Status & Control Register 2 . . . B-9
B.2.4.5 Status Register 3 . . . B-9
B.3 P.O.S.T. LED . . . B-10

C. AD VP2/001-10 Breakout . . . C-1

C.1 Introduction . . . C-1
C.2 Layout . . . C-1
C.3 Header Configuration . . . C-2

D. AD VP2/001-20 Breakout . . . D-1

D.1 Introduction . . . D-1
D.2 Layout . . . D-1
D.3 Header Configuration . . . D-2

E. AD VP2/002-10 Breakout . . . E-1

E.1 Introduction . . . E-1
E.2 Layout . . . E-1
E.3 Header Configuration . . . E-2

F. AD VP2/003-10 Breakout . . . F-1

F.1 Introduction . . . F-1
F.2 Layout . . . F-1
F.3 Header and Connector Configuration . . . F-2

G. AD VP2/003-20 Breakout . . . G-1

G.1 Introduction . . . G-1
G.2 Layout . . . G-1
G.3 Header and Connector Configuration . . . G-2

H. AD VP2/003-30 Breakout . . . H-1

H.1 Introduction . . . H-1
H.2 Layout . . . H-1
H.3 Header and Connector Configuration . . . H-2

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

Table of Figures

Figure 1-1	Overview	1-1
Figure 3-1	Factory Jumper Settings	3-2
Figure 3-2	Battery Fitting and CMOS CLEAR Jumper	3-3
Figure 3-3	SCSI Jumpers	3-4
Figure 3-4	Console Mode Jumper	3-5
Figure 3-5	VSA Mode Jumper	3-6
Figure 3-6	Factory Test Jumper	3-7
Figure 3-7	Watchdog Jumper	3-8
Figure 3-8	Front Panel Switch Jumper	3-9
Figure 3-9	VME System Reset Jumper	3-10
Figure 5-1	Overview	5-1
Figure 5-2	PCI Configuration Registers	5-2
Figure 5-3	Memory Map	5-3
Figure 5-4	I/O Space Map	5-5
Figure 5-5	Front Panel LEDs	5-11
Figure 6-1	The Main Setup Menu	6-4
Figure 6-2	Primary Master / Primary Slave Sub-Menu	6-6
Figure 6-3	Memory Cache Sub-Menu	6-8
Figure 6-4	Advanced Setup Menu	6-10
Figure 6-5	Advanced Chipset Control Sub-Menu	6-11
Figure 6-6	I/O Device Configuration Sub-Menu	6-12
Figure 6-7	Security Setup Menu	6-13
Figure 6-8	Universe Setup Menu	6-14
Figure 6-9	PCI Slave Images Sub-Menu	6-16
Figure 6-10	VME Slave Images Sub-Menu	6-17
Figure 6-11	Boot Setup Menu	6-18
Figure 6-12	Boot Device Priority Sub-Menu	6-19
Figure 6-13	Hard Drive Sub-Menu	6-20
Figure 6-14	Exit Menu	6-21
Figure 6-15	SCSI BIOS Main Menu	6-26
Figure 6-16	SCSI BIOS Change Adapter Status Menu	6-26
Figure 6-17	SCSI BIOS Adapter Boot Order Menu	6-27
Figure 6-18	SCSI BIOS Additional Adapter Configuration Menu	6-27
Figure 6-19	SCSI BIOS Utilities Menu	6-28
Figure 6-20	SCSI BIOS Adapter Setup Menu	6-28
Figure 6-21	SCSI BIOS Device Selections Menu	6-31
Figure 6-22	Device Options Menu	6-31
Figure A-1	Connector Layout	A-2
Figure A-2	Front Panel Connector Layout	A-2
Figure A-3	Keyboard Connector	A-8
Figure A-4	Mouse Connector	A-9
Figure A-5	RJ-45 Connector (Front View)	A-14
Figure A-6	USB Connector	A-16
Figure B-1	Real Mode Memory Map	B-1
Figure B-2	Protected Mode Memory Map	B-2
Figure B-3	I/O Space Map	B-2
Figure C-1	AD VP2/001-10 P2 Breakout Connectors	C-1
Figure D-1	AD VP2/001-20 P2 Breakout Connectors	D-1
Figure E-1	AD VP2/002-10 P2 Breakout Connectors	E-1
Figure F-1	AD VP2/003-10 P2 Breakout Connectors	F-1
Figure G-1	AD VP2/003-20 P2 Breakout Connectors	G-1
Figure H-1	AD VP2/003-30 P2 Breakout Connectors	H-1

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

Table of Tables

Table 1-1 VME P2 Breakout Interfaces · · · · · 1-4
Table 5-1 Interrupt Assignments · · · · · 5-6
Table 5-2 PCI Arbiter Priority Assignment · · · · · 5-10
Table 6-1 Main Setup Options · · · · · 6-4
Table 6-2 Primary Master / Primary Slave Options · · · · · 6-7
Table 6-3 Memory Cache Options · · · · · 6-8
Table 6-3 Memory Cache Options (continued) · · · · · 6-9
Table 6-4 Advanced Setup Options · · · · · 6-10
Table 6-5 Advanced Chipset Control Options · · · · · 6-11
Table 6-6 I/O Device Configuration Options · · · · · 6-12
Table 6-7 Security Setup Options · · · · · 6-13
Table 6-8 Universe Configuration Options · · · · · 6-15
Table 6-9 PCI Slave Images Options · · · · · 6-16
Table 6-10 VME Slave Images Options · · · · · 6-17
Table 6-11 Boot Options · · · · · 6-18
Table 6-12 Exit and Save Options · · · · · 6-21
Table 6-13 POST Flash Codes · · · · · 6-22
Table A-1 Serial Interface Pin-outs - Channel 0 (COM1) · · · · · A-3
Table A-2 VME Interface Pin-outs · · · · · A-4
Table A-3 P2 Connector Pin-outs (Wide SCSI, Panel Link and EIDE) · · · · · A-5
Table A-4 P2 Connector Pin-outs (Narrow SCSI, Printer and EIDE) · · · · · A-6
Table A-5 P2 Connector Pin-outs (EIDE Only) · · · · · A-7
Table A-6 Keyboard Connector Pin Assignments · · · · · A-8
Table A-7 Mouse Connector Pin Assignments · · · · · A-9
Table A-8 PMC Carrier P9 Connector · · · · · A-10
Table A-9 PMC Carrier P10 Connector · · · · · A-11
Table A-10 PMC J1 Connector · · · · · A-12
Table A-11 PMC J2 Connector · · · · · A-13
Table A-12 RJ-45 Connector Pin-outs · · · · · A-14
Table A-13 30-way Debug Connector Pin-outs · · · · · A-15
Table A-14 USB Pin Assignments · · · · · A-16
Table A-15 VGA Pin Assignments · · · · · A-17
Table B-1 PCI Device IDs · · · · · B-2
Table B-2 I/O Address Map · · · · · B-3
Table B-3 Interrupt Assignments · · · · · B-3
Table C-1 SCSI 50-wayIDC Header · · · · · C-2
Table C-2 Floppy 34-way IDC Header · · · · · C-2
Table C-3 Printer 26-way IDC Header · · · · · C-3
Table C-4 External Reset/NMI 3-way Header · · · · · C-3
Table C-5 5V 3-way Header · · · · · C-3
Table D-1 EIDE 40-way IDC Header · · · · · D-2
Table D-2 Floppy 34-way IDC Header · · · · · D-2
Table D-3 Printer 26-way IDC Header · · · · · D-3
Table D-4 External Reset/NMI 3-way Header · · · · · D-3
Table D-5 5V 3-way Header · · · · · D-3
Table E-1 SCSI 50-way IDC Header · · · · · E-2
Table E-2 EIDE 40-way IDC Header · · · · · E-3
Table E-3 Floppy 34-way IDC Header · · · · · E-3
Table E-4 Printer 26-way IDC Header · · · · · E-4
Table E-5 USB Connector · · · · · E-5
Table E-6 USB 5-pin Header · · · · · E-5
Table E-7 External Reset/NMI 3-way Header · · · · · E-5
Table E-8 5V 3-way Header · · · · · E-5
Table E-9 General purpose I/O · · · · · E-5
Table F-1 Narrow SCSI 50-way IDC Header · · · · · F-2
Table F-2 EIDE 40-way IDC Header · · · · · F-3

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

Table F-3	Floppy 34-way IDC Header	F-3
Table F-4	USB Connector	F-4
Table F-5	Alternate USB Header	F-4
Table F-6	External Reset/NMI 3-way Header	F-4
Table F-7	General Purpose I/O	F-4
Table F-8	Wide SCSI 68-way Connector	F-5
Table F-9	DFP Connector	F-6
Table G-1	EIDE 40-way IDC Header	G-2
Table G-2	Floppy 34-way IDC Header	G-2
Table G-3	DFP Connector	G-3
Table H-1	Narrow SCSI 50-way IDC Header	H-2
Table H-2	Floppy 34-way IDC Header	H-2
Table H-3	External Reset/NMI 3-way Header	H-3
Table H-4	Wide SCSI 68-way Connector	H-3
Table H-5	DFP Connector	H-4

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

Introduction and Overview

1.1 General

This manual is a guide and reference handbook for engineers and system integrators who wish to use the Concurrent Technologies' VP PSE/P3x ultra high-performance single board computer. The board has been designed for high-speed multiprocessing applications using a PC-AT™ architecture operating in a VME bus environment.

The VP PSE/P3x board is available in several different variants which differ by the speed and type of the CPU, SCSI and/or EIDE interfaces and the amount of fitted SDRAM. Currently the board is available with 433MHz or 566MHz Celeron, or 700MHz or 850MHz Embedded Pentium III processors, the product names being VP PSE/C12, VP PSE/C13, VP PSE/P33 and VP PSE/P34 respectively. The boards may be supplied with one of a range of SDRAM sizes, as specified by a two-digit suffix to the board name, refer to the product data sheet for further details. This suffix also identifies which combination of SCSI, EIDE and Printer interfaces are provided via the board's P2 connector. Further details of these options are given in section 1.3. References to the board in this document will use the name VP PSE/P3x unless they apply only to a specific variant, in which case the full name will be used.

The information contained in this manual has been written to provide users with all the information necessary to configure, install and use the VP PSE/P3x as part of a system. It assumes that the user is familiar with the VME bus and PC-AT bus architectures and features, and also that the user has access to component catalogs for detailed programming information on the peripheral integrated circuits used on the board.

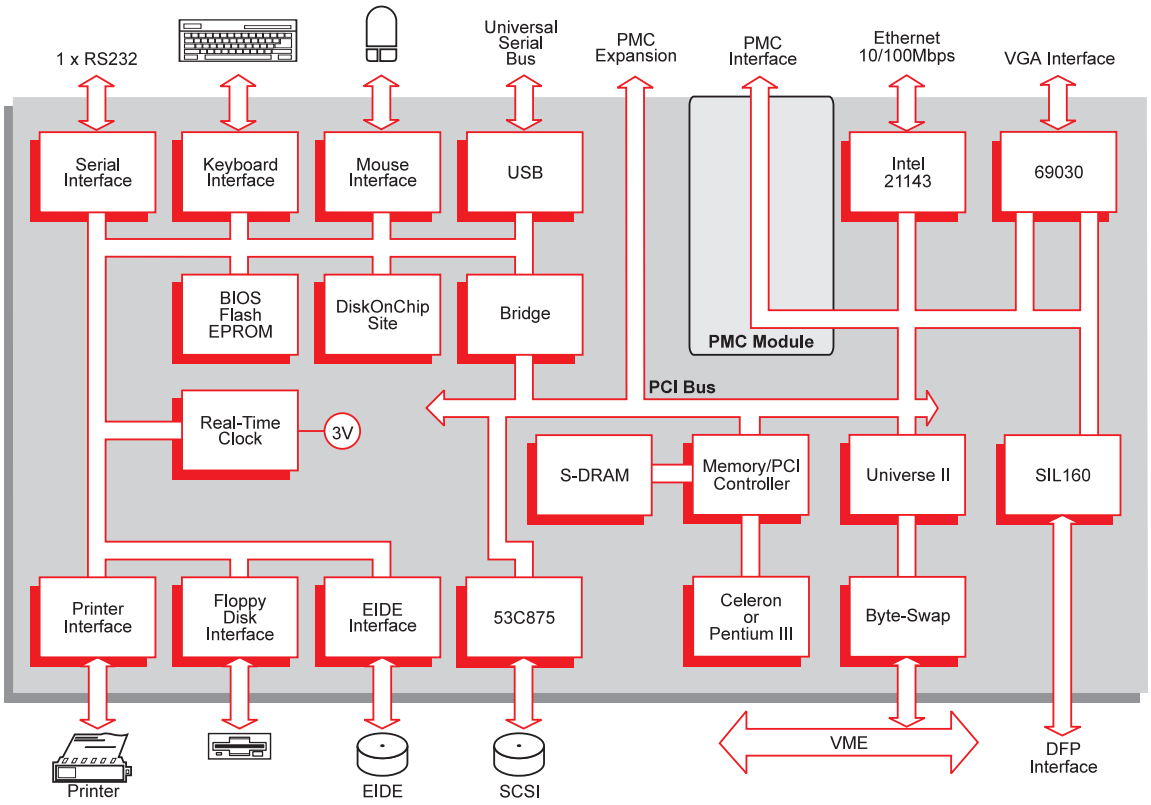


Figure 1-1 Overview

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

1.2 The VP PSE/P3x - Main Features

The VP PSE/P3x is a member of the Concurrent Technologies range of single-board computers for the VME bus architecture. It has been designed as a powerful single board computer based upon the Celeron or Pentium III processor, the 53C875 SCSI Controller, the 21143 Ethernet™ controller, the 69030 Graphics Controller, the IEEE P1386.1 PMC interface and standard PC-AT based peripherals.

1.2.1 Central Processor

The 64-bit central processor used on this board may be one of two types: a very high performance low power Celeron microprocessor, operating internally at either 433 or 566MHz, or an ultra high performance low power Pentium III microprocessor, operating internally at either 700 or 850MHz, as defined by the product order code. The processor supports the Dual Independent Bus (DIB) architecture with the backside bus connected to the on die Level 2 cache operating at the core frequency and the frontside bus connected to the memory controller at 66MHz on Celeron processors, or 100MHz on Pentium III processors. The processor is capable of addressing 4 Gbytes of physical memory all of which is cacheable and 64 Terabytes of virtual memory. Both processor types are upwardly code-compatible with the other members of the x86 family of microprocessors.

The processor has an in-built floating point coprocessor for compatibility with 486 and 386/387 designs.

1.2.2 Memory Controller

The memory controller is implemented with Intel's 443BX device which is part of the 440BX chipset. The chipset provides the SDRAM controller, CPU to PCI accesses and PCI master accesses to SDRAM. The chipset supports concurrent CPU and PCI bus operations. Processor burst and pipelining modes are supported to achieve a transfer rate of up to 211Mbytes/s (Celeron) or 355 Mbytes/s (Pentium III) from SDRAM.

1.2.3 Level 2 Cache

A 64-bit wide 128 Kbyte (Celeron) or 256 Kbyte (Pentium III) Level 2 cache operating at the core frequency is supported. The cache is implemented on the processor die. On board SDRAM is the only memory that can be cached by the processor's Level 1 and/or Level 2 cache.

1.2.4 SDRAM

The board supports both soldered on SDRAM devices to provide 64 or 128 Mbytes and a single 144 pin SODIMM which can be a 8M x 64 bit, a 16M x 64 bit, or a 32M x 64 bit, SDRAM module. A maximum of 384 Mbytes SDRAM may be fitted to the board.

1.2.5 PCI Bus and Bridges

A 32-bit wide PCI bus provides a high performance, up to 132 Mbytes/s, connection between the memory controller, SCSI controller, Ethernet controller, Graphics Controller, PCI expansion, PC-AT peripherals and bridge to VME.

The PCI Expansion interface supports the Concurrent Technologies AD SBC/PMC PMC carrier board allowing additional I/O expansion using PMC modules.

1.2.6 EPROM

The board has 512Kbytes of PC BIOS Flash EPROM arranged as 8 bit wide.

1.2.7 Serial Communications

The VP PSE/P3x has a single RS232 serial data communication channel. This is accessed through a 9-way D-type connector on the board's front panel. The RS232 channel is implemented by the Super I/O Controller which contains two 16550 compatible Serial Communications Controllers. The source clock used for baud rate generation is 24MHz.

1.2.8 SCSI Controller

The VP PSE/P3x has a Ultra Small Computer System Interface (UltraSCSI) implemented with a 53C875 Wide SCSI Controller which connects directly to the PCI bus. The SCSI Interface is available via the P2 connector as an 16-bit (wide) single-ended interface.

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

1.2.9 EIDE Controller

The VP PSE/P3x has an EIDE/Ultra DMA33 interface implemented with an Intel® PIIX4E PCI to ISA bridge device which connects directly to the PCI Bus. The EIDE interface is available via the P2 connector.

1.2.10 Ethernet Controller

The Intel 21143 Ethernet controller and Integrated Circuit Systems ICS 1892 physical interface are used to provide a high performance PCI to Ethernet interface. A 10Base-T or 100 Base-TX Cat5 RJ45 interface is provided via the front panel. This interface can be set to run at 10 or 100 Mbits per second.

1.2.11 Graphics Controller

The Intel CHIPS 69030 is used to provide a high performance graphics accelerator with 4 Mbytes of integrated memory. A CRT interface is provided via a standard 15-way high density connector on the front panel. A DFP flat panel interface is provided via P2.

1.2.12 VME Interface

The VP PSE/P3x is VME64 compatible. It supports A32, A24, and A16 addressing. It supports D64, D32, D16 and D08 transfers and MBLT, BLT, ADOH, RMW and LOCK cycles. The interface is implemented using the 91C142 UNIVERSE II™ PCI to VME bridge which provides master and slave capability. Fast hardware byte swapping is supported, under software control, for 16 and 32-bit aligned single cycle and block transfers.

1.2.13 Keyboard & Mouse

PS/2™ type keyboard and mouse interfaces are available via the front panel.

1.2.14 Real Time Clock

A battery backed RTC device provides PC-AT clock, calendar and configuration RAM functions. The RTC and BIOS are year 2000 compliant.

1.2.15 Floppy Disk

A floppy disk interface is provided by the Super I/O Controller for up to two floppy drives and is connected via the P2 connector.

1.2.16 Parallel Interface

The VP PSE/P3x has a single parallel port interface made available at the P2 connector, implemented by the Super I/O Controller chip.

1.2.17 USB

A single USB interface is provided via the front panel . A second USB interface is made available at the P2 connector. The interfaces are capable of running at 1.5 and/or 12 Mbit/s.

1.2.18 DiskOnChip® Site

A 32 pin DIL socket is provided for the user to install a DiskOnChip device. Any capacity of device may be installed, up to the current maximum of 144 Mbytes.

1.2.19 PMC Interface

A PMC interface which supports single width PMC modules which complies with the IEEE P1361.1 standard is provided. A 5V PCI signaling environment is supported.

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

1.3 Additional Board Options

The VP PSE/P3x board may be ordered in one of two hard-wired configurations, and with one of several different VME P2 connector breakout or adaptor modules.

If ordered in an EIDE configuration the SCSI interface is not available, and the AD VP2/001-20 breakout is used (see Appendix D). If ordered in a SCSI configuration, the narrow SCSI bus is accessible through the pins in the 96-way VME P2 connector, and the EIDE interface is available through the outer rows of pins in a 160-way VME P2 connector. Appendix C describes the AD VP2/001-10 breakout module which uses a 96-way VME P2 connector and does not provide an EIDE interface.

Appendix E describes the AD VP2/002-10 breakout module which provides both narrow SCSI and EIDE interfaces, as well as a printer interface and uses a 160-way VME P2 connector.

Appendix F describes the AD VP2/003-10 breakout module which can be used with the board in the SCSI configuration. This breakout module uses a 160-way VME P2 connector and provides both Wide SCSI and EIDE interfaces, as well as a DFP interface.

Appendix G describes the ADVP2/003-20 breakout module which provides and EIDE interface and a DFP interface through a 96-way VME P2 connector.

Appendix H describes the AD VP2/003-30 breakout module which provides a Wide SCSI interface through a 96-way VME P2 connector.

Table 1-1 summarizes the interfaces available using each of these VME P2 breakouts.

Breakout	VME P2 Connector Pins	Wide SCSI	Narrow SCSI	EIDE	Floppy	Printer	DFP	USB	EXT Reset
AD VP2/001-10	96		✓		✓	✓			✓
AD VP2/001-20	96			✓	✓	✓			✓
AD VP2/002-10	160		✓	✓	✓	✓		✓	✓
AD VP2/003-10	160	✓	✓	✓	✓		✓	✓	✓
AD VP2/003-20	96			✓	✓		✓		
AD VP2/003-30	96	✓	✓		✓		✓		✓

Table 1-1 VME P2 Breakout Interfaces

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

Installation

2.1 General

This chapter contains general information on unpacking and inspecting the VP PSE/P3x after shipment, installation and powering up the board.

CAUTION It is strongly advised that, when handling the VP PSE/P3x and its associated components, the user should at all times wear an earthing strap to prevent damage to the board as a result of electrostatic discharge.

2.2 Unpacking and Inspection

Immediately after the board is delivered to the user's premises the user should carry out a thorough inspection of the package for any damage caused by negligent handling in transit.

CAUTION If the packaging is badly damaged or water-stained the user must insist on the carrier's agent being present when the board is unpacked.

Once unpacked, the board should be inspected carefully for physical damage, loose components etc. In the event of the board arriving at the customer's premises in an obviously damaged condition Concurrent Technologies or its authorized agent should be notified immediately.

2.3 Working Environment

The board's operating environment needs to be carefully controlled if it is to function with maximum reliability. The power, cooling and humidity requirements are set out in Appendix A.

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

2.4 Installation and Power-up

Once the VP PSE/P3x has been configured to the user's requirements (see Chapter 3) it can be installed in the system backplane.

The board is installed and powered up as follows:

1. Make sure that system power is turned OFF.
2. If your system requires the use of the EMC spring contact strips provided, fit strips into the slots on the long edges of the front panel.
3. Slide the board into the designated slot, making sure that the board fits neatly into the runners.
4. Push the board into the card-cage until the P1 and P2 connectors are firmly located.
5. Screw the ejector handle retaining bolts into the holes in the chassis.
6. Connect up the I/O cables to the connectors on the board's front panel and fix in place with the connectors' retaining screws.
7. If using the P2 breakout, install on the rear P2 connector and connect up the I/O cables.
8. Power-up the system. The following sequence of events should then occur:
   a) The green "RUN" LED and the yellow "POST" LED on the front panel will light up.
   b) The yellow "POST" LED will go out.

   If power-up does not follow the sequence described above this will indicate that the board is not operational.

NOTE This sequence of events assumes the VP PSE/P3x has Concurrent Technologies standard BIOS firmware and that the board is configured to the factory setting described in Chapter 3 (Configuration).

Refer to Chapter 5, section 5.6 for a complete description of the front panel indicator LEDs.

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

Configuration

3.1 General

The VP PSE/P3x board offers the user a range of optional hardware and operating configurations. This chapter details the jumper settings and component options available. Board functionality is not described here in detail, and it is recommended that any user unfamiliar with the VP PSE/P3x should read Chapter 5, Functional Description, before attempting to alter the existing board configuration.

CAUTION It is strongly advised that, when handling the VP PSE/P3x and its associated components, the user should wear an earthing strap to prevent damage to the board as a result of electrostatic discharge.

3.2 Factory Setting

When delivered to the customer the VP PSE/P3x will be configured as follows:

Processor type	• Celeron or Pentium III processor fitted as defined by the product order code. Front-side bus speed set to 66MHz or 100MHz as appropriate for the processor type.
Memory	• BIOS Flash EPROM has a single 29F040 byte wide device fitted as standard. SDRAM will be up to 384 Mbytes as defined by the product order code.
PCI Expansion	• No carrier module fitted.
SCSI	• On board termination enabled, with board supplying termination power.
Jumpers	• Console/graphics - set to graphics • CMOS clear - disabled • Watchdog - disabled • VSA Mode - disabled • Factory Test - off • VME System Reset - reset board with VME reset

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

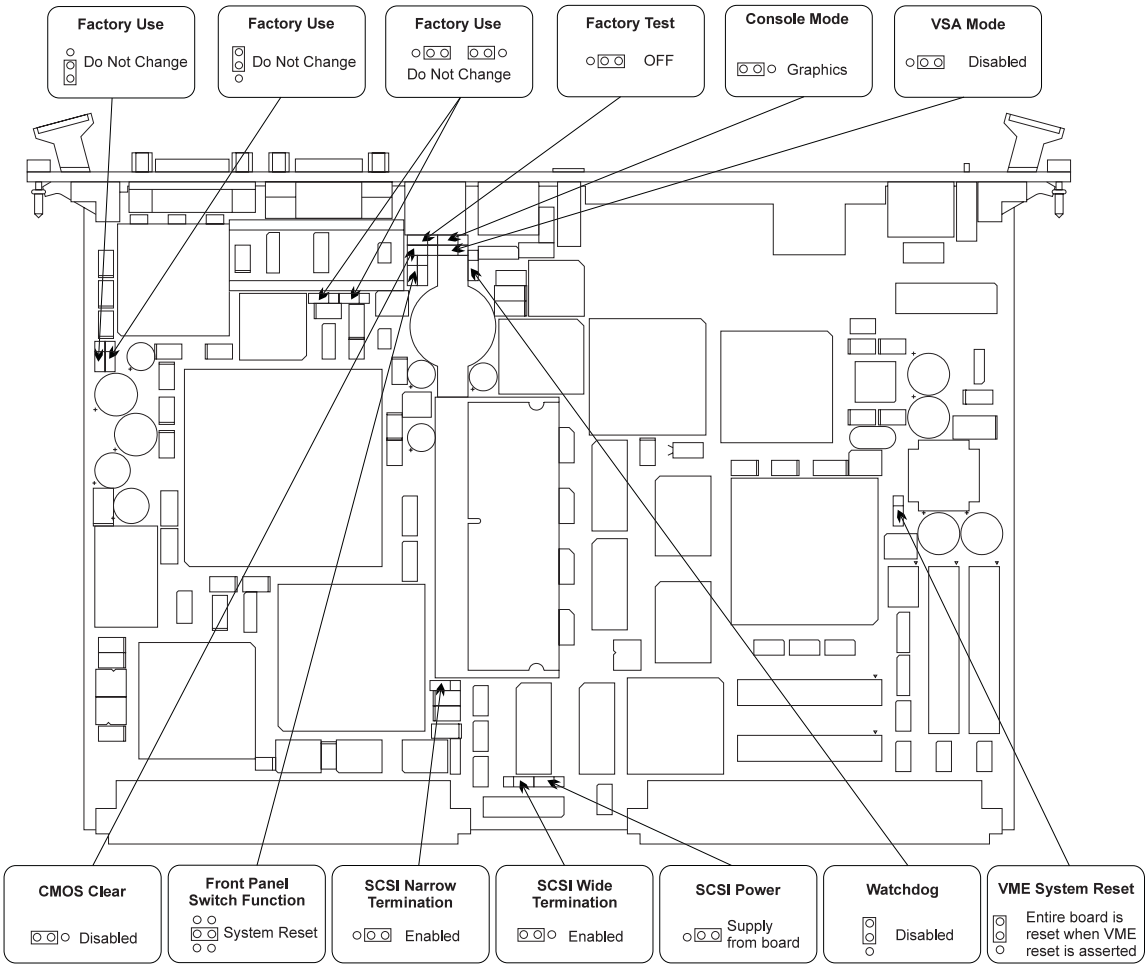


Figure 3-1 Factory Jumper Settings

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

3.3 Battery Installation/Replacement

Power to the PIIX4E device, which provides clock/calendar functions, is sourced from the on board 3V Lithium battery. The battery will need to be replaced when its voltage falls below 2.6 Volts.

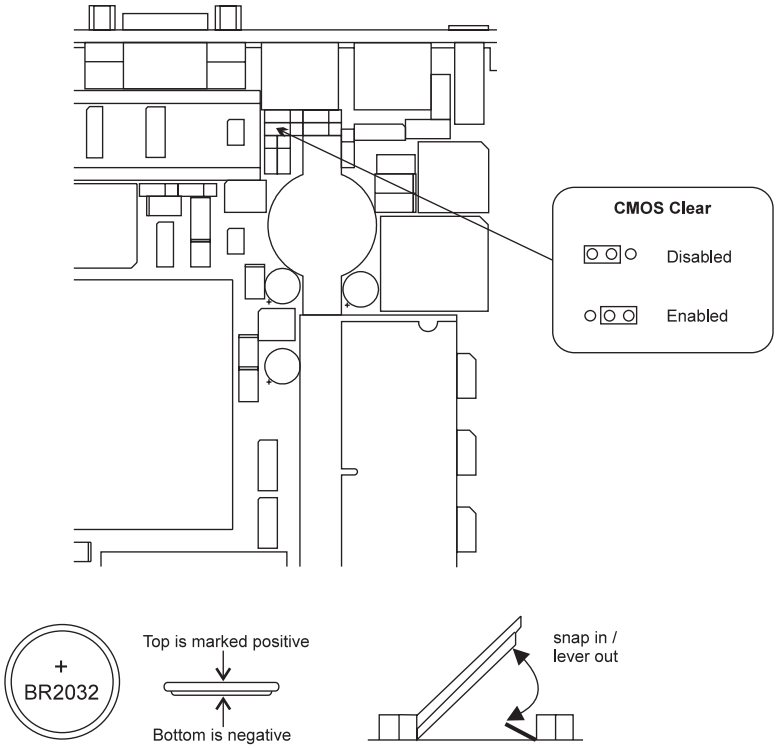


Figure 3-2 Battery Fitting and CMOS CLEAR Jumper

NOTE Setting the CMOS CLEAR jumper to enabled, momentarily, after battery is fitted and before applying power to the board, will clear the CMOS RAM to a known state and is recommended whenever the battery is replaced/installed for the first time.

CAUTION When replacing the battery, proper anti-static precautions must be observed.

WARNING Dispose of battery properly. DO NOT BURN. The date and time settings will need to be initialized if the battery is disconnected.

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

3.4 SCSI Interface Functions

There are three sets of jumpers associated with the SCSI interface which are shown in Figure 3-3.

SCSI NARROW TERMINATION jumper. This jumper determines whether the on board active bus terminates for data bits D0 through D7 and SCSI signals are enabled. SCSI power must be present to enable the terminators.

SCSI WIDE TERMINATION jumper. This jumper determines whether the on board active SCSI bus terminators for data bits D8 through D15 and DP1 (parity) are enabled. SCSI power must be present to enable the terminators. If the board is used on the narrow SCSI bus the SCSI WIDE TERMINATION jumper should be set to enabled to prevent the unused signals from floating.

SCSI POWER jumper. The power for the on-board active SCSI terminators can either be sourced from the SCSI bus or the board itself depending upon the configuration of the SCSI POWER jumper. When set to provide power from the board itself, power is available on the SCSI bus for powering the termination devices at the other end of the bus as well.

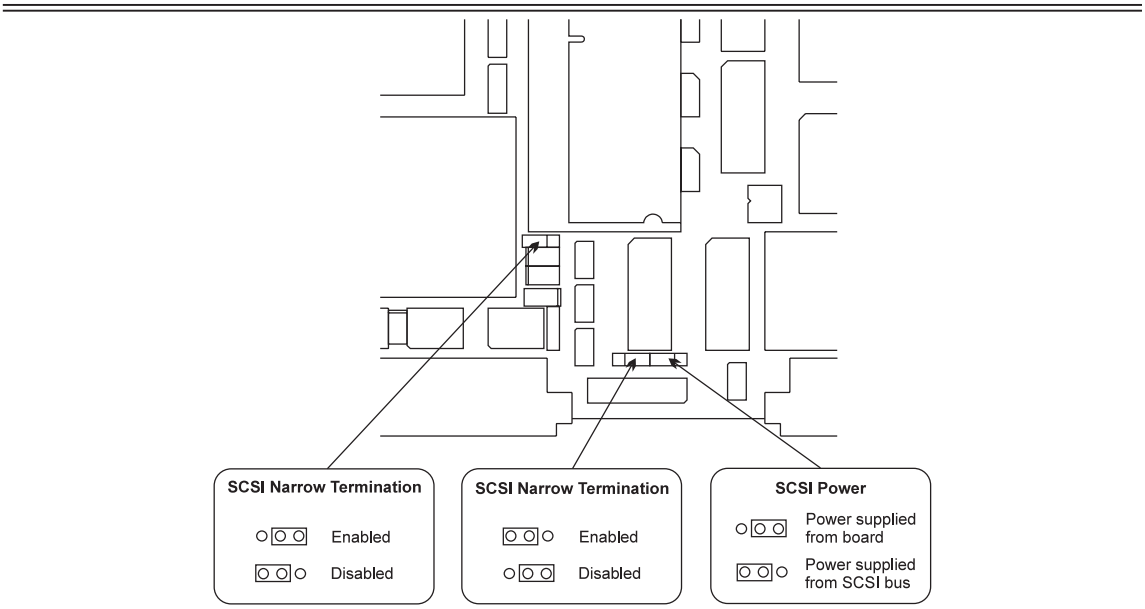


Figure 3-3 SCSI Jumpers

NOTE Power from the board to the SCSI terminations is provided via a self resetting 1A fuse. If excessive power is taken due to a fault condition, e.g. cable short to GND, the fuse will limit the current to a very low level. After the fault has been removed the fuse will revert back to a low impedance state without any user intervention.

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

3.5 Console Mode

The user can select a serial console or graphics mode of operation for the PC BIOS.

When console is selected, standard input and output is redirected to serial channel 0 (COM1). When graphics is selected, standard input is via the keyboard and standard output is via the VGA interface.

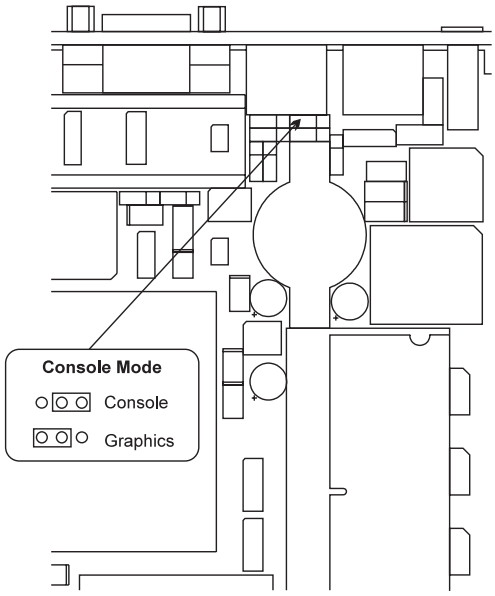


Figure 3-4 Console Mode Jumper

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

3.6 VSA Mode Jumper

The VSA Mode jumper determines the way in which the board starts operating - either VSA Mode or PC BIOS Mode.

Further details of VSA Mode operation are given in Chapter 7, and of PC BIOS mode operation in Chapter 6.

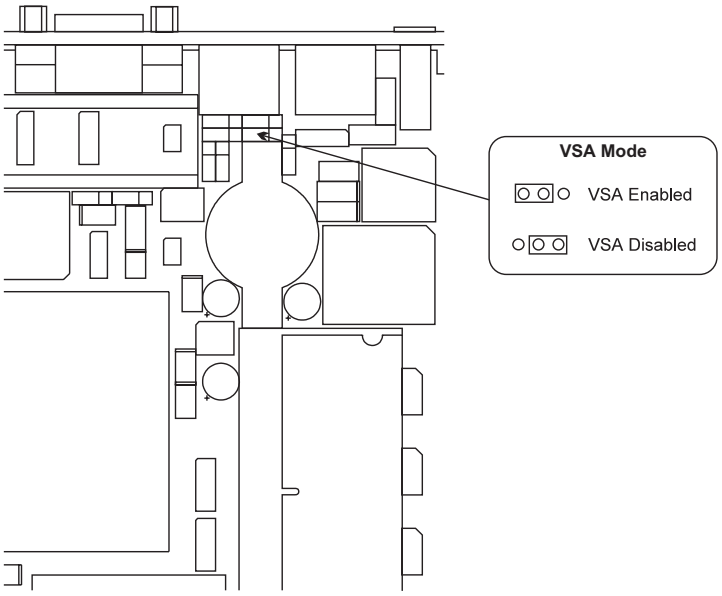


Figure 3-5 VSA Mode Jumper

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

3.7 Test Jumpers

A FACTORY TEST jumper is defined, and is reserved for use during factory testing of the board.

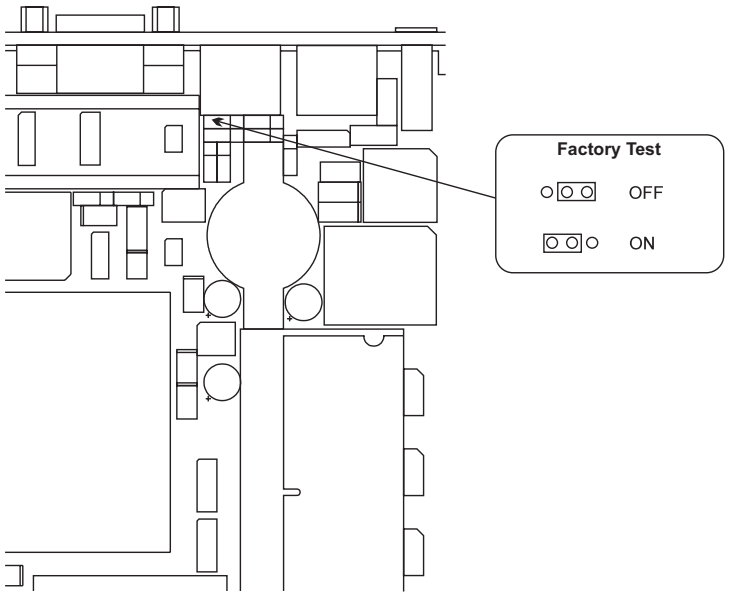


Figure 3-6 Factory Test Jumper

WARNING Changing the FACTORY TEST jumper will affect the power-on sequence of the BIOS and may prevent the board from bootloading its operating software.

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

3.8 Watchdog Jumper

The watchdog jumper provides an overriding enable/disable to the watchdog circuit. When in the disable position the watchdog will be disabled regardless of any software settings. In the enable position the watchdog is under software control, as described in Section B.2.4.3.

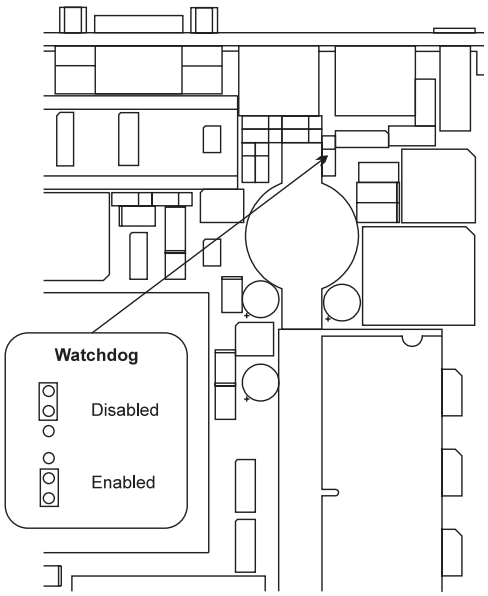


Figure 3-7 Watchdog Jumper

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

3.9 Front Panel Switch Jumper

The Front Panel switch can be configured to cause a local Non-Maskable Interrupt (NMI), a local CPU reset, or a system wide reset, as shown in Figure 3-8.

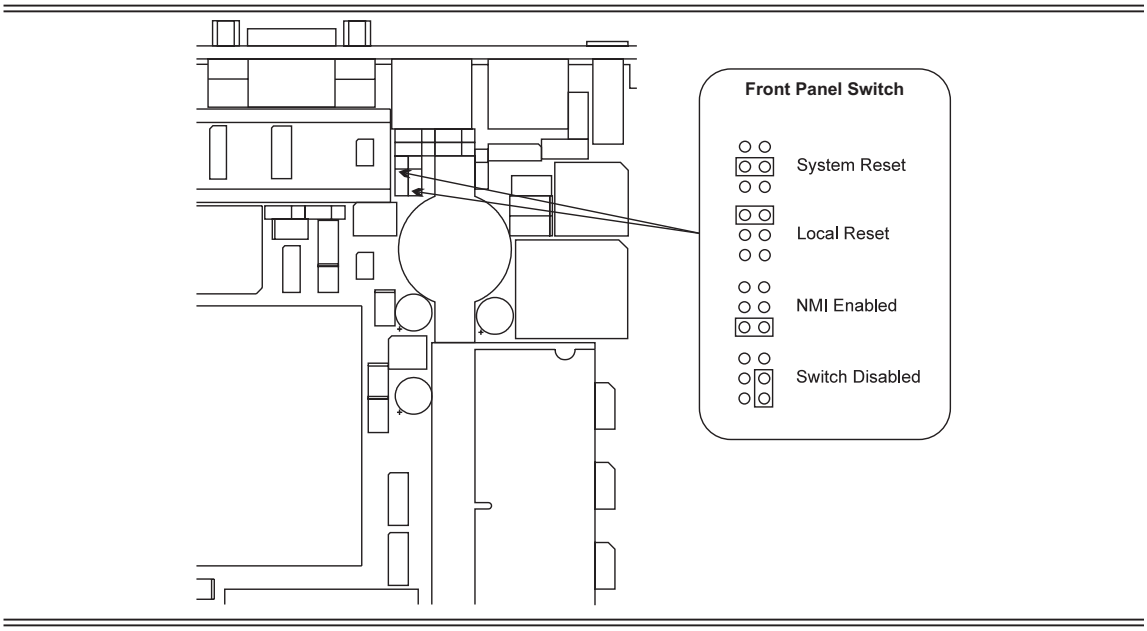


Figure 3-8 Front Panel Switch Jumper

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

3.10 VME System Reset Jumper

The VME reset jumper controls how VME bus reset affects the VP PSE/P3x board. This provides the option of having the board reset independently of or together with a VME Bus reset.

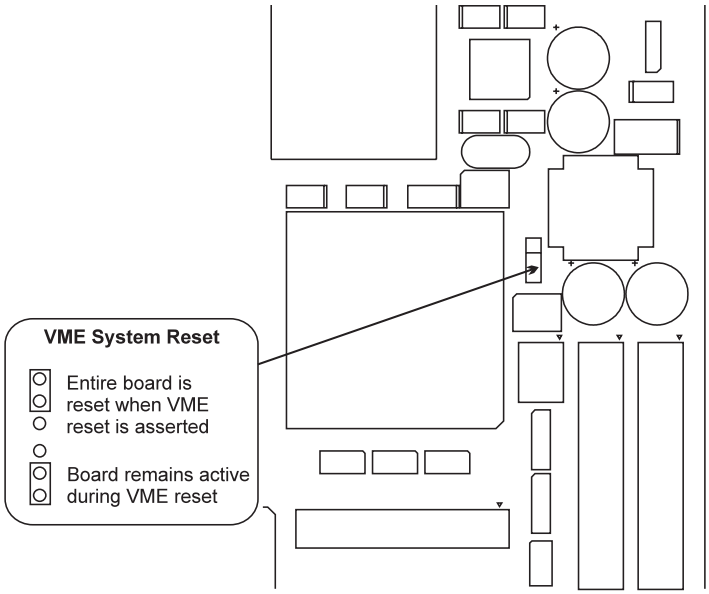


Figure 3-9 VME System Reset Jumper

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

Programming Examples

This chapter contains a number of example programming sequences, written in assembler and C-code, illustrating how to control some of the hardware sub-systems provided by the board.

The VP PSE/P3x board has an IBM PC-AT compatible architecture, therefore this chapter will cover only the hardware sub-systems which are additional to those found on standard PCs.

4.1 PCI Bus Interface

The following source code comprises a C header file and associated functions which provide a basic interface to PCI bus configuration registers on any device on the board. These functions are not required for simple I/O or memory accesses, which are handled transparently by the board hardware, once the board has been correctly configured.

```
extern void PciRead (unsigned char bus,
                        unsigned char device,
                        unsigned char func,
                        unsigned char reg,
                        unsigned char * pBuf,
                        unsigned char unsigned chars);

extern void PciWrite (unsigned char bus,
                        unsigned char device,
                        unsigned char func,
                        unsigned char reg,
                        unsigned char * pBuf,
                        unsigned char unsigned chars);

extern void PciReadInfo (unsigned char bus,
                            unsigned char device,
                            unsigned short * pVendorId,
                            unsigned short * pModuleId,
                            unsigned char * pRevision);

extern unsigned char PciBusScan (unsigned char bus,
                                    unsigned short vendorId,
                                    unsigned short moduleId,
                                    unsigned char * pDevice);

/* PCI bus definitions */

#define PCI_MAX_DEVICE      (unsigned char)16

#define PCI_BUS_0           (unsigned char)0
#define PCI_FUNC_0          (unsigned char)0

/* PCI configuration register definitions */

#define PCI_VID             (unsigned char)0x00     /* Vendor ID offset */
#define PCI_MID             (unsigned char)0x02     /* Module ID offset */
#define PCI_COMMAND         (unsigned char)0x04
#define PCI_ENABLE_IO       (unsigned short)0x0001
#define PCI_ENABLE_MEMORY   (unsigned short)0x0002
#define PCI_ENABLE_MASTERING (unsigned short)0x0004
#define PCI_ENABLE_SERR     (unsigned short)0x0100
#define PCI_STATUS          (unsigned char)0x06
#define PCI_RID             (unsigned char)0x08
#define PCI_PROGRAMMING_IF  (unsigned char)0x09
#define PCI_SUB_CLASS_CODE (unsigned char)0x0a
```


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

```
#define PCI_CLASS_CODE      (unsigned char)0x0b
#define PCI_CACHE_LINE_SIZE (unsigned char)0x0c
#define PCI_LATENCY_TIMER   (unsigned char)0x0d
#define PCI_HEADER_TYPE     (unsigned char)0x0e
#define PCI_BIST            (unsigned char)0x0f
#define PCI_BASE_ADDR_0     (unsigned char)0x10
#define PCI_BASE_ADDR_1     (unsigned char)0x14
#define PCI_BASE_ADDR_2     (unsigned char)0x18
#define PCI_BASE_ADDR_3     (unsigned char)0x1c
#define PCI_BASE_ADDR_4     (unsigned char)0x20

#define PCI_BASE_ADDR_5     (unsigned char)0x24

#define PCI_ROM_BASE        (unsigned char)0x30

#define PCI_INTERRUPT_LINE  (unsigned char)0x3c
#define PCI_INTERRUPT_PIN   (unsigned char)0x3d
#define PCI_MIN_GNT         (unsigned char)0x3e
#define PCI_MAX_LAT         (unsigned char)0x3f



#define CONFADD             (unsigned short)0x0cf8 /* PCI config address port */
#define CONFDATA            (unsigned short)0x0cfc /* PCI config data port */


/*****************************************************************************
 * PciReadReg(...)
 *
 * This function accepts the following parameters:
 *  PCI bus number
 *  PCI device number
 *  PCI function number
 *  PCI register number
 */

static unsigned long PciReadReg
(
    unsigned char bus,
    unsigned char device,
    unsigned char func,
    unsigned char reg
)
{
unsigned long regAddress;
unsigned long result;

    /* Set configuration address from given parameters */

    regAddress = (0x80000000 |   /* Enable config access */
                (((unsigned long)bus) << 16) |          /* PCI bus number */
                (((unsigned long)func) << 8) |          /* Function number */
                (((unsigned long)device) << 11) |       /* Device number */
                 ((unsigned long)reg & 0x0000000fc));   /* Register offset */

    sysOutDword (CONFADD, regAddress); /* Set up config reg address */

    /* Read the full 32-bit result unmodified */

    result = sysInDword (CONFDATA);

    return (result);
}
```


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

```
/*****************************************************************************
 * PciWriteReg(...)
 *
 * This function accepts the following parameters:
 *  PCI bus number
 *  PCI device number
 *  PCI function number
 *  PCI register number
 *  Value to be written
 */

static void PciWriteReg
(
    unsigned char bus,
    unsigned char device,
    unsigned char func,
    unsigned char reg,
    unsigned long value
)
{
unsigned long regAddress;

    /* Set configuration address from given parameters */

    regAddress = (0x80000000 | /* Enable config access */
                (((unsigned long)bus) << 16) |          /* PCI bus number */
                (((unsigned long)func) << 8) |          /* Function number */
                (((unsigned long)device) << 11) |       /* Device number.  */
                 ((unsigned long)reg & 0x0000000fc));   /* Register offset */

    sysOutDword (CONFADD, regAddress);  /* Set up config reg address */

    /* Write out value unmodified */

    sysOutDword (CONFDATA, value);

    return;
}


/*****************************************************************************
 * PciRead(...)
 * Read a set of char values from PCI configuration space, using long accesses
 *
 * This function accepts the following parameters:
 *  PCI bus number
 *  PCI device number
 *  PCI function number
 *  First PCI register
 *  Pointer to set of register values
 *  Number of chars in register value set
 */

void PciRead
(
    unsigned char   bus,
    unsigned char   device,
    unsigned char   func,
    unsigned char   reg,
    unsigned char*  pbuf,
    unsigned char   chars
)
{
unsigned long value;
unsigned char oddchars;
```


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

```
    /* Read all the chars one long (or partial long) at a time */

    while (chars > 0)
    {
        value = PciReadReg (bus, device, func, reg);
        value >>= (8 * (reg & 0x03));
        oddchars = 4 - (reg & 0x03);

        if (oddchars > chars)
            oddchars = chars;

        while (oddchars > 0)
        {
            *pbuf++ = value & 0xff;
            value >>= 8;
            reg++;
            chars--;
            oddchars--;
        }
    }

    return;
}


/*****************************************************************************
 * PciWrite(...)
 * Write a set of char values to PCI configuration space, using long accesses
 *
 * This function accepts the following parameters:
 *  PCI bus number
 *  PCI device number
 *  PCI function number
 *  First PCI register number
 *  Pointer to set of register values
 *  Number of chars in set of register values
 */

void PciWrite
(
    unsigned char   bus,
    unsigned char   device,
    unsigned char   func,
    unsigned char   reg,
    unsigned char*  pbuf,
    unsigned char   chars
)
{
unsigned long value;
unsigned long mask;
unsigned char width;
unsigned char shift;

    /* Write all the chars one long (or partial long) at a time */

    while (chars > 0)
    {
        shift = 8 * (reg & 0x03);
        width = 4 - (reg & 0x03);

        if (width > chars)
            width = chars;

        if (width >= 4)
```


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

```
        {
            value = *(unsigned long *)pbuf;
            PciWriteReg (bus, device, func, reg, value);
        }

        else
        {
            mask = (1 << (8 * width)) - 1;
            mask <<= shift;
            value = ((*(unsigned long *)pbuf) << shift) & mask;

            PciWriteReg (bus, device, func, reg,
                (PciReadReg (bus, device, func, reg) & ~mask) | value);
        }

        pbuf  += width;
        reg   += width;
        chars -= width;
    }

    return;
}


/*****************************************************************************
 * PciReadInfo(...)
 * Reads the vendor id, device id and the revision number from the config
 * space and writes the results into the passed pointers
 *
 * Returns:
 *  The Vendor ID register value
 *  The Module ID register value
 *  The Module Revision register value
 */

void PciReadInfo
(
    unsigned char   bus,
    unsigned char   device,
    unsigned short* pVendorId,
    unsigned short* pModuleId,
    unsigned char*  pRevision
)
{
    PciRead (bus, device, PCI_FUNC_0, PCI_VID, (unsigned char *)pVendorId,
                (unsigned char)2);
    PciRead (bus, device, PCI_FUNC_0, PCI_MID, (unsigned char *)pModuleId,
                (unsigned char)2);
    PciRead (bus, device, PCI_FUNC_0, PCI_RID, (unsigned char *)pRevision,
                (unsigned char)1);

    return;
}
```


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

```
/*****************************************************************************
 * PciBusScan(...)
 * Locates a module on the selected PCI bus from its vendor & module IDs
 * The scan begins at the PCI bus slot selected by the device ID parameter
 *
 * Returns:
 *  The device ID of the module on the selected bus
 */

unsigned char PciBusScan
(
    unsigned char   bus,
    unsigned short  vendorId,
    unsigned short  moduleId,
    unsigned char*  pDevice
)
{
unsigned short scanVendorId;
unsigned short scanModuleId;
unsigned char scanRevision;
unsigned char found;

    found = FALSE;

    while (*pDevice < PCI_MAX_DEVICE)
    {
        PciReadInfo (bus, *pDevice, &scanVendorId, &scanModuleId,
                        &scanRevision);

        if ((scanVendorId == vendorId) && (scanModuleId == moduleId))
        {
            found = TRUE;
            break;
        }

        (*pDevice)++;
    }

    return found;
}
```


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

4.2 Enabling the Control/Status Registers

The Control and Status registers are enabled by Device 9 Generic Decode Chip-select, which is controlled through a device resource register belonging to the Power Management function of the PIIX4.

```
/* Port base IO address */

#define CONTROL_REGS        0x210

/* PCI device numbers */

#define PIIX4_DEVICE        7        /* PCI device number for PIIX4 */

/* PCI configuration space registers */

#define INTEL_PIIX4_REG62   0x62    /* device resource reg */


void vEnableControlStatusRegs (void)
{
UINT32  dGpBase     /* general purpose registers - I/O base address */
UINT8   bTemp;      /* temporary BYTE */
UINT8   wTemp;      /* temporary WORD */

    /* Write the control register base address */

    pci_write_reg (0, PIIX4_DEVICE, 3, INTEL_PIIX4_REG60, PCI_WORD,
                   CONTROL_REGS);

    /* Enable PCS0 */

    bTemp = (UINT8)pci_read_reg (0, PIIX4_DEVICE, 3, INTEL_PIIX4_REG62,
                                 PCI_BYTE);

    pci_write_reg (0, PIIX4_DEVICE, 3, INTEL_PIIX4_REG62, PCI_BYTE,
                       bTemp | 0xC0);

} /* vEnableControlStatusRegs () */
```


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

4.3 Programming the Watchdog

The following functions show how to use the watchdog facility available through the status and control registers.

It is worth noting that the Software Enable bit in the Watchdog Status and Control register does not read back the value last written; it is the output from the error checking logic, so it will not read back as enabled until two restarts have been performed (equally it will not display disabled until two further restarts have been performed).

```
/* Status and control registers */

#define STATCTL_BASE        0x0210

#define WATCHDOG_STATCTL    (STATCTL_BASE + 4)
#define     WD_PAT_MASK         0x03
#define     WD_PAT_1            0x02
#define     WD_PAT_2            0x01
#define     WD_ACTION_MASK      0x04
#define     WD_ACTION_NMI       0x00
#define     WD_ACTION_RST       0x04
#define     WD_STATUS           0x08
#define     WD_HW_DISABLE       0x10
#define     WD_SW_ENABLE        0x20


/*****************************************************************************
 * vEnableWatchdog: configure the watchdog for NMI or RESET on timeout
 *
 * RETURNS: none
 */

void vEnableWatchdog
(
    UINT8   bTimeoutAction  /* action on timeout: NMI or Reset */
)
{
    UINT8   bTemp;


    bTemp = inbyte (WATCHDOG_STATCTL);

    bTemp &= ~WD_ACTION_MASK;   /* set watchdog timeout action */
    bTemp |= bTimeoutAction;

    bTemp |= WD_SW_ENABLE;      /* software enable of watchdog */
    bTemp &= ~WD_PAT_MASK;      /* clear the pat bits */

    outbyte (WATCHDOG_STATCTL, bTemp | WD_PAT_1);   /* set and pat twice */
    outbyte (WATCHDOG_STATCTL, bTemp | WD_PAT_2);
    outbyte (WATCHDOG_STATCTL, bTemp | WD_PAT_1);

} /* vEnableWatchdog () */


/*****************************************************************************
 * vDisableWatchdog: de-configure the watchdog and disable
 *
 * RETURNS: none
 */

void vDisableWatchdog (void)
{
    UINT8   bTemp;
```


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

```
    bTemp = inbyte (WATCHDOG_STATCTL);

    bTemp &= ~WD_ACTION_MASK;       /* set watchdog action to NMI */
    bTemp |= WD_ACTION_NMI;

    bTemp &= ~WD_SW_ENABLE;         /* software disable the watchdog */

    outbyte (WATCHDOG_STATCTL, bTemp | WD_PAT_1);   /* set and pat twice */
    outbyte (WATCHDOG_STATCTL, bTemp | WD_PAT_2);
    outbyte (WATCHDOG_STATCTL, bTemp | WD_PAT_1);

} /* vDisableWatchdog () */


/*****************************************************************************
 * vRestartWatchdog: restart the watchdog to prevent timeout
 *
 * Bits 1:0 of the Watchdog status / control register are inverted before
 * writing back.
 *
 * RETURNS: none
 */

void vRestartWatchdog (void)
{
    UINT8   bTemp;
    UINT8   bPat;

    bTemp = inbyte (WATCHDOG_STATCTL);

    bPat = (~bTemp) & WD_PAT_MASK;  /* get complement of bits 1:0 */
    bTemp &= ~WD_PAT_MASK;          /* clear the bits 1:0 in data */
    bTemp |= bPat;                  /* set new bits */

    outbyte  (WATCHDOG_STATCTL, bTemp);

} /* vRestartWatchdog () */
```


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

4.4 Intel (Digital) 21143 Ethernet Controller

The code fragments below give examples of programming the Intel (Digital) 21143 Ethernet controller for loopback testing. These examples show how to initialize the device, prepare data descriptors and buffers, send initialization frames and drive the receive and transmit engines. Code is also given for reading and writing the configuration EEPROM.

```
/* defines */

 /* general defines */

#define RING_ADDR       0x00040000
#define RING_SIZE       0x00020000  /* 128kBytes */
#define TX_RING_OS      0x00000000
#define RX_RING_OS      0x00010000

#define TX_MSG_SIZE     512                 /* data message size */
#define DATA_SIZE       1500                /* size of data buffer */
#define F_BUFFER_SIZE   (DATA_SIZE + 18)    /* frame buff: data + extra */
#define NUM_BUFFERS     4                   /* number of buffers */

#define INIT_FRAME_SIZE (16 * 12)           /* 16 entries, each 3 DWORDs */
#define SROM_SIZE       0x400               /* 1024 bytes */

#define LB_NONE         0
#define LB_INTERNAL     1
#define LB_EXTERNAL     2

#define PT_MII          0
#define PT_10BASET      1
#define PT_AUI          2
#define PT_SYM          3

#define SP_10           0
#define SP_100          1

#define DIGITAL_VID     0x1011
#define _21143_DID      0x0019

/* file scope globals */

static UINT32   dDeviceIoBase;  /* I/O base address of registers */
static UINT8    bDeviceVector;  /* interrupt vector for device */


/*****************************************************************************
 * wEthLoopbackTest: handles, internal, external and network loopback tests.
 *
 * RETURN: E_OK or an E__... error code.
 */

UINT16 wEthLoopbackTest
(
    tagTxRing* pasTxRing,   /* pointer to TX ring elements */
    tagRxRing* pasRxRing,   /* pointer to RX ring elements */
    UINT8 bMode,            /* loopback mode */
    UINT8 bNumBuffers       /* transfer size (complete buffers) */
)
{
UINT32  dDeviceState;   /* status of the device, TX and RX processes */
UINT32  dTxState;       /* status of the TX process only */
UINT32  dRxState;       /* status of the RX process only */
UINT32  dCrc;           /* frame's CRC */
UINT16  wTestStatus;    /* error reporting */
```


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

```
UINT16  wTimeout;       /* transmission timeout counter */
UINT8   abStationId[6]; /* placeholder for station address */
char    achBuffer[80];  /* text formatting buffer */

    vCreateRings (pasTxRing, pasRxRing); /* build ring structure */

    vGetStationAddress (abStationId); /* get this stations address */

    /* Set receive addresses via setup frame */

    wTestStatus = wSendSetupFrame (pasTxRing, &abStationId[0]);

    if (wTestStatus != E_OK)
        return (wTestStatus);

    /* Prepare data buffers */

    vZeroRxBuffers (pasRxRing);
    vSetTxBuffers (pasTxRing, &abStationId[0], &abStationId[0], bNumBuffers);

    vCsrWrite (CSR4_TX_BASE, RING_ADDR + TX_RING_OS); /* reset ring pointers */
    vCsrWrite (CSR3_RX_BASE, RING_ADDR + RX_RING_OS);

    /* Select loopback mode. Start Tx/Rx process */

    vCsrClear (CSR6_OP_MODE, CSR6__OM);
    vCsrSet (CSR6_OP_MODE, ((bMode << OM_SHIFT)));

    vSetBufferOwn (pasTxRing, pasRxRing); /* pass ownership to 21143 */
    vStartRxTx (); /* GO */

    /* Wait for completion of transmission: TX suspended (due to underrun) */

    wTimeout = 1000; /* 1 second */

    do {
        dDeviceState = dCsrRead (CSR5_STATUS);

        dTxState = (dDeviceState & CSR5__TS) >> TS_SHIFT;
        dRxState = (dDeviceState & CSR5__RS) >> RS_SHIFT;

        vGeneralDelay (1); /* 1mS delay */
        wTimeout--;

    } while ((wTimeout > 0) && ((dTxState != TS_SUSPEND) ||
                                (dRxState == RS_WAIT_RX)));

    /* Check for errors on TX side first, then RX, then compare data */

    wTestStatus = wStopTxRx ();

    if (wTimeout == 0) /* handle timeout errors before STOP errors */
    {
        if (dTxState == 0)
            wTestStatus = E__TX_TIMEOUT;

        else
            wTestStatus = E__RX_TIMEOUT;
    }

    else if (wTestStatus == E_OK) /* Ok, so check status and data buffer */
    {
        /* Check device status first */

        if ((dDeviceState & CSR5__FBE) != 0)
```


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

```
            wTestStatus = E__PCI_BUS;

        if (wTestStatus == E_OK)
            wTestStatus |= wTestRxBuffers (pasTxRing, pasRxRing, bNumBuffers);

        dCrc = dComputeCrc (pasTxRing->pbTxBuff, F_BUFFER_SIZE - 4);

        if (dCrc != *((UINT32*)(pasRxRing->pbRxBuff + F_BUFFER_SIZE - 4)))
            return (E__CRC_ERROR);
    }

    return (wTestStatus);

} /* wEthLoopbackTest () */


/*****************************************************************************
 * vCreateRings: This function prepares the TX and RX descriptor ring and
 * initializes the TX and RX buffers. The buffers are linked to descriptors,
 * which are then linked together. Pointers to each created descriptor and
 * buffer are stored in an array of structures; one for RX, one for TX.
 *
 * RETURNS: none (pointers returned via passed parameters)
 */

void vCreateRings
(
    tagTxRing* pasTxRing, /* pointer to TX ring elements: RETURNED */
    tagRxRing* pasRxRing /* pointer to RX ring elements: RETURNED */
)
{
selector    selRingPool;    /* selector for allocated memory */
tagTxDesc*  psThisTxDesc;   /* pointer to the current TX descriptor */
tagRxDesc*  psThisRxDesc;   /* pointer to the current RX descriptor */
UINT32  dRingPool;          /* absolute address of free memory area */
UINT32  dCurrentOffset;     /* used to track offset into free memory */
UINT8*  pbThisBuffer;       /* pointer to a TX or RX buffer for init */
UINT8   bIndex;             /* counter for indexing descriptors/buffers */

    dRingPool = RING_ADDR;          /* base address for ring creation */
    dCurrentOffset = TX_RING_OS;    /* offset from the base of pool */

    /* Create selector for access to the descriptors and buffers */

    selRingPool = set_scratch_1 (dRingPool, RING_SIZE); /* 128kBytes */

    /* Initialize the transmit descriptors and buffers */

    for (bIndex = 0; bIndex < NUM_BUFFERS; bIndex++)
    {
        /* Create pointers to a descriptor and a buffer in the ring pool */

        psThisTxDesc = (tagTxDesc*) buildptr (selRingPool,
                                (void near*)dCurrentOffset);
        dCurrentOffset += sizeof (tagTxDesc);

        pbThisBuffer = (UINT8*) buildptr (selRingPool,
                                        (void near*)dCurrentOffset);

        /* Set buffer address and the chain address */

        psThisTxDesc->dBuffAddr = dRingPool + dCurrentOffset;
        dCurrentOffset += F_BUFFER_SIZE;

        psThisTxDesc->dChain = RING_ADDR + dCurrentOffset;
```


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

```
            /* Clear descriptor status and set control bits for chained structure */

            psThisTxDesc->dStatus = 0;
            psThisTxDesc->dControl = TDES1__LS | /* last segment */
                                     TDES1__FS | /* also first */
                                     TDES1__TCH | /* chained */
                                     F_BUFFER_SIZE;

            if (bIndex == NUM_BUFFERS-1)
                psThisTxDesc->dControl |= TDES1__TER;

            /* Enter pointers into returned structures */

            (pasTxRing + bIndex)->psTxDesc = psThisTxDesc;
            (pasTxRing + bIndex)->pbTxBuff = pbThisBuffer;
        }

        /* Initialize the receive descriptors and buffers */

        dCurrentOffset = RX_RING_OS; /* offset from the base of pool */

        for (bIndex = 0; bIndex < NUM_BUFFERS; bIndex++)
        {
            /* Create pointers to a descriptor and a buffer */

            psThisRxDesc = (tagRxDesc*) buildptr (selRingPool,
                                                  (void near*)dCurrentOffset);
            dCurrentOffset += sizeof (tagRxDesc);

            pbThisBuffer = (UINT8*) buildptr (selRingPool,
                                              (void near*)dCurrentOffset);

            /* Set buffer address and the chain address */

            psThisRxDesc->dBuffAddr = dRingPool + dCurrentOffset;
            dCurrentOffset += F_BUFFER_SIZE;

            psThisRxDesc->dChain = RING_ADDR + dCurrentOffset;

            /* Clear descriptor status and set control bits for chained structure */

            psThisRxDesc->dStatus = 0;
            psThisRxDesc->dControl = RDES1__RCH | /* chained */
                                     F_BUFFER_SIZE;

            if (bIndex == NUM_BUFFERS-1)
                psThisRxDesc->dControl |= RDES1__RER;

            /* Enter pointers into returned structures */

            (pasRxRing + bIndex)->psRxDesc = psThisRxDesc;
            (pasRxRing + bIndex)->pbRxBuff = pbThisBuffer;
        }

    } /* vCreateRing () */
```


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

```
/*****************************************************************************
 * wSendSetupFrame: create setup frame and send to the controller. The setup
 * frame contains the station addresses that the controller receive channel
 * will respond to.
 *
 * RETURNS: E_OK or an E_... error code.
 */

UINT16 wSendSetupFrame
(
    tagTxRing*  pasTxRing,      /* pointer to TX ring elements */
    UINT8*      abStationId     /* station address for RX */
)
{
tagTxDesc*  psTxDesc;   /* pointer to the current TX descriptor */
UINT32  dSavedTxCtl;    /* saved state of TX descriptors control field */
UINT32  dTxState;       /* status of the TX process, CSR6 */
UINT16  wTimeout;       /* timeout counter */
UINT16  wTestStatus;    /* error reporting */
UINT16  wOffset;        /* offset into setup buffer */
UINT8*  pbBuffAddr;     /* pointer to a transmit buffer */
int     iEntry;         /* index into the station address list */
char    achBuffer[80];  /* text formatting buffer */

    wTestStatus = E_OK;

    /* Prepare an init. frame in the first buffer */

    psTxDesc = pasTxRing->psTxDesc;
    pbBuffAddr = pasTxRing->pbTxBuff;

    dSavedTxCtl = psTxDesc->dControl;

    /* First entry is the broadcast address */

    *(pbBuffAddr + 0) = 0xFF; /* DWORD #1 */
    *(pbBuffAddr + 1) = 0xFF;
    *(pbBuffAddr + 4) = 0xFF; /* DWORD #2 */
    *(pbBuffAddr + 5) = 0xFF;
    *(pbBuffAddr + 8) = 0xFF; /* DWORD #3 */
    *(pbBuffAddr + 9) = 0xFF;

    wOffset = 12;

    /* All other entries are copies of the station address */

    for (iEntry = 0; iEntry < 16; iEntry++) /* 16 copies of station address */
    {
        *(pbBuffAddr + (wOffset + 0)) = abStationId[0]; /* DWORD #1 */
        *(pbBuffAddr + (wOffset + 1)) = abStationId[1];
        *(pbBuffAddr + (wOffset + 4)) = abStationId[2]; /* DWORD #2 */
        *(pbBuffAddr + (wOffset + 5)) = abStationId[3];
        *(pbBuffAddr + (wOffset + 8)) = abStationId[4]; /* DWORD #3 */
        *(pbBuffAddr + (wOffset + 9)) = abStationId[5];

        wOffset += 12; /* 3*DWORD */
    }

    /* Enable the transmitter for these frames, pause while processed */

    psTxDesc->dControl = TDES1__TCH | TDES1__SET | INIT_FRAME_SIZE;
    psTxDesc->dStatus = TDES0__OWN;

    vCsrSet (CSR6_OP_MODE, CSR6__ST); /* start transmitter */
```


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

```
    /* Wait for frame to be read by 21143, ie. OWN bit cleared */

    wTimeout = 100;

    while ( ((psTxDesc->dStatus & TDES0__OWN) != 0) && (wTimeout != 0) )
    {
        vGeneralDelay (1);
        wTimeout--;
    }

    /* Stop transmitter and check for errors. Restore previous descriptor
     * state only if no errors detected
     */

    wTestStatus = wStopTxRx ();

    if (wTimeout == 0)
        wTestStatus = E__STOP_FAIL;

    else if (wTestStatus == E_OK)
        psTxDesc->dControl = dSavedTxCtl; /* Restore saved data and exit */

    return (wTestStatus);

} /* wSendSetupFrame () */


/*****************************************************************************
 * vSetTxBuffers: this function sets the content of a specified number of
 * transmit buffers in the descriptor ring. A ramp pattern is used.
 *
 * RETURNS: none
 */

void vSetTxBuffers
(
    tagTxRing* pasTxRing,   /* pointer to TX ring elements */
    UINT8* abSourceId,      /* source station address */
    UINT8* abDestId,        /* destination station address */
    UINT8   bNumBuffers     /* number of frames to set */
)
{
tagTxDesc*  psTxDesc;   /* pointer to the current TX descriptor */
UINT16  wCntr;          /* buffer fill counter */
UINT8*  Index;          /* counter for indexing descriptors/buffers */

    /* Put data into our transmit buffers, zero remainder */

    for (bIndex = 0; bIndex < NUM_BUFFERS; bIndex++)
    {
        psTxDesc = (pasTxRing + bIndex)->psTxDesc;
        pbTxBuff = (pasTxRing + bIndex)->pbTxBuff;

        /* Insert destination and source addresses plus length for new frames */

        memcpy (pbTxBuff, abDestId, 6); /* destination */
        memcpy (pbTxBuff + 6, abSourceId, 6); /* source */

        *(pbTxBuff + 12) = 0x0E; /* data size LB (512 + 6+6+2) */
        *(pbTxBuff + 13) = 0x02; /* data size HB (512 + 6+6+2) */

        pbTxBuff += 14; /* move pointer to data part of this buffer */

        /* Put test data into buffers and fill in control register for this
         * descriptor
```


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

```
             */

            if (bIndex < bNumBuffers)
            {
                for (wCntr = 0; wCntr < DATA_SIZE; wCntr++)
                    *(pbTxBuff + wCntr) = (UINT8)wCntr;

                psTxDesc->dControl &= ~(TDES1__TER | TDES1__TBS1 | TDES1__TBS2);
                psTxDesc->dControl |= (DATA_SIZE + 6+6+2);
                psTxDesc->dControl |= TDES1__FS; /* first segment of frame */
                psTxDesc->dControl |= TDES1__LS; /* last segment of frame */

                if (bIndex == (bNumBuffers - 1))
                    psTxDesc->dControl |= TDES1__TER; /* also last descriptor */
            }

            else
            {
                for (wCntr = 0; wCntr < DATA_SIZE; wCntr++)
                    *(pbTxBuff + wCntr) = 0;

                psTxDesc->dControl &= ~(TDES1__TER | TDES1__TBS1 | TDES1__TBS2);
                psTxDesc->dControl |= TDES1__FS; /* first segment of frame */
                psTxDesc->dControl |= TDES1__LS; /* last segment of frame */
            }
        }

    } /* vSetTxBuffers () */


    /*************************************************************************
     * vSetBufferOwn: this function sets the ownership bits for both TX and RX
     * buffers in their respective descriptor ringa.
     *
     * RETURNS: none
     */

    void vSetBufferOwn
    (
        tagTxRing*  pasTxRing,  /* pointer to TX ring elements */
        tagRxRing*  pasRxRing   /* pointer to RX ring elements */
    )
    {
    tagRxDesc*  psRxDesc;   /* pointer to the current RX descriptor */
    tagTxDesc*  psTxDesc;   /* pointer to the current TX descriptor */
    int     iIndex;         /* counter for indexing descriptors */

        /* Set ownership bit in all descriptors - passes ownership to 21143.
         * This is done in reverse order to prevent race conditions
         */

        for (iIndex = NUM_BUFFERS - 1; iIndex >= 0 ; iIndex--)
        {
            psTxDesc = (pasTxRing + iIndex)->psTxDesc;
            psRxDesc = (pasRxRing + iIndex)->psRxDesc;

            psTxDesc->dStatus = TDES0__OWN;
            psRxDesc->dStatus = RDES0__OWN;
        }

    } /* vSetBufferOwn () */


    /*************************************************************************
     * vZeroRxBuffers: this function zeros the contents of the receive buffers.
```


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

```
 *
 * RETURNS: none
 */

void vZeroRxBuffers
(
    tagRxRing* pasRxRing /* pointer to RX ring elements */
)
{
UINT16  wCntr;      /* buffer index counter */
UINT8*  pbBuffAddr; /* pointer to a receive buffer */
UINT8   bIndex;     /* counter for indexing descriptors/buffers */

    /* Zero all our buffers */

    for (bIndex = 0; bIndex < NUM_BUFFERS; bIndex++)
    {
        pbBuffAddr = (pasRxRing + bIndex)->pbRxBuff;

        for (wCntr = 0; wCntr < F_BUFFER_SIZE; wCntr++)
        {
            *(pbBuffAddr++) = 0;
        }
    }

} /* vZeroRxBuffers () */


/*****************************************************************************
 * wPciInitialize: locate and initialize PCI configuration registers for this
 * device instance.
 *
 * RETURN: E_OK or E__... error code.
 */

UINT16 wPciInitialize (void)
{
UINT16  wFound;         /* result of the device search */
UINT16  wTemp;          /* temporary word value */
UINT8   bBus;           /* PCI address of device */
UINT8   bFunc;          /* -"- */
UINT8   bDev;           /* -"- */
UINT8   bIntIrq;        /* IRQ number for interrupt signal */
char    achBuffer[80];  /* text formatting buffer */

    /* First, locate this instance of the card */

    wFound = PciFindDevice (&bBus, &bDev, &bFunc, DIGITAL_VID, _21143_DID,
                            bEthInstanceCtr);

    if (wFound == DEVICE_NOT_FOUND)
        return (E__CARD_NOT_FOUND);

    /* Enable all PCI accesses to the device; not supporting memory
     * write and invalidate cycles, PERR or SERR.
     */

    wTemp = pci_read_reg (bBus, bDev, bFunc, PCI_COMMAND, PCI_WORD);
    wTemp = PCI_BMEN | PCI_IOEN;
    pci_write_reg (bBus, bDev, bFunc, PCI_COMMAND, PCI_WORD, wTemp);

    /* Reset the status register - write 1's for all used bits */

    pci_write_reg (bBus, bDev, bFunc, PCI_STATUS, PCI_WORD, 0xF780);
```


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

```
    /* Set the latency timer - use value suggested by Linux driver code */

    pci_write_reg (bBus, bDev, bFunc, PCI_LATENCY_TIMER, PCI_BYTE, 0x64);

    /* Determine the I/O base address assigned to this device */

    dDeviceIoBase = pci_read_reg (bBus, bDev, bFunc, PCI_IO_BASE, PCI_DWORD);
    dDeviceIoBase &= 0xFFFFFFFE; /* strip I/O bit */

    /* Get interrupt vector */

    bIntIrq = pci_read_reg (bBus, bDev, bFunc, PCI_INT_LINE, PCI_BYTE);

    bDeviceVector = irq_to_vector (bIntIrq);

    /* Finally! take controller out of sleep mode to run mode */

    pci_write_reg (bBus, bDev, bFunc, 0x40, PCI_DWORD, 0L);

    return (E_OK); /* card configured */

} /* wPciInitialize () */


/*****************************************************************************
 * vInitializeDevice: initialize the device registers for this instance.
 *
 * RETURN: none
 */

void vInitializeDevice (void)
{
    /* Set port to MII/SYM */

    vCsrWrite (CSR6_OP_MODE, CSR6__SCR | CSR6__PCS | CSR6__MBO | CSR6__PS);

    /* Software reset */

    vCsrWrite (CSR0_BUS_MODE, 0);
    vGeneralDelay (1);
    vCsrWrite (CSR0_BUS_MODE, CSR0__SWR);
    vGeneralDelay (1);
    vCsrWrite (CSR0_BUS_MODE, 0);

    /* Set registers */

    vCsrWrite (CSR0_BUS_MODE, 0x00009000); /* default operating mode */

    vCsrWrite (CSR7_INT_EN, 0); /* no interrupts enabled */

    vCsrWrite (CSR3_RX_BASE, RING_ADDR + RX_RING_OS);
    vCsrWrite (CSR4_TX_BASE, RING_ADDR + TX_RING_OS);

    vCsrWrite (CSR13_SIA_CON, 0);
    vCsrWrite (CSR14_SIA_TXRX, 0);

    /* Programme LED & General purpose IO */

    vCsrWrite (CSR15_SIA_PORT, 0x08010008);
    vCsrWrite (CSR15_SIA_PORT, 0x00000008);
    vGeneralDelay(2); /* hold external device reset */
    vCsrWrite (CSR15_SIA_PORT, 0x00010008); /* release reset */

    vCsrSet (CSR6_OP_MODE, CSR6__MBO | CSR6__PS );
    vCsrSet (CSR6_OP_MODE, CSR6__MBO | CSR6__TR );
```


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

```
    vCsrWrite (CSR10_ROM_ADDR, 0x000001FF); /* initialize to 1's */
    vCsrWrite (CSR10_ROM_ADDR, 0); /* initialize to zero */

    vCsrWrite (CSR5_STATUS, 0x0C01FFFF); /* write with 1's to clear */
    vCsrWrite (CSR9_ROM_CTL, 0); /* deselect all */
    vCsrWrite (CSR11_TIMER, 0); /* initialize to zero */
    vCsrWrite (CSR12_SIA_STAT, 0xFFFFFF0F); /* clear all */

} /* vInitializeDevice () */


/*******************************************************************************
 * vStartRxTx: starts both receive and transmit processes.
 *
 * RETURNS: none
 */

void vStartRxTx (void)
{
    /* Start receiver */

    vCsrSet (CSR6_OP_MODE, CSR6__SR);

    /* A short delay */

    vGeneralDelay (500);

    /* Start the transmitter */

    vCsrSet (CSR6_OP_MODE, CSR6__ST);

    /* Prod the device to poll for descriptors */

    vCsrWrite (CSR1_TX_POLL, 1);

} /* vStartRxTx () */


/*******************************************************************************
 * wStopTxRx: stops both transmit and receive processes.
 *
 * RETURNS: E_OK or E__... code if STOP failed.
 */

UINT16 wStopTxRx (void)
{
UINT32  dDeviceState;   /* general device status */
UINT32  dTxState;       /* status of the transmit process */
UINT32  dRxState;       /* status of the receive process */
UINT16  wTestStatus;    /* error reporting */
UINT16  wTimeout;       /* timeout counter */
int     iIndex;         /* descriptor index */
char    achBuffer[80];  /* text formatting buffer */

    wTestStatus = E_OK;

    /* Issue stop to transmitter, then receiver */

    vCsrClear (CSR6_OP_MODE, CSR6__ST);
    vCsrClear (CSR6_OP_MODE, CSR6__SR);

    /* Poll transmit and receive states: wait until actually stopped */

    wTimeout = 1000; /* 1 second timeout */
```


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

```
    do {
        dDeviceState = dCsrRead (CSR5_STATUS);

        dTxState = (dDeviceState & CSR5__TS) >> TS_SHIFT;
        dRxState = (dDeviceState & CSR5__RS) >> RS_SHIFT;

        wTimeout--;
        vGeneralDelay (1);

    } while ((wTimeout > 0) && (dTxState != TS_STOP) && (dRxState != RS_STOP));

    if (wTimeout == 0)
        wTestStatus = E__STOP_FAIL;

    return (wTestStatus);

} /* wStopTxRx () */


/*****************************************************************************
 * vGetStationAddress: this function reads the controller's station address
 * into the supplied buffer.
 *
 * RETURNS: none - address returned via passed pointer.
 */

void vGetStationAddress
(
    UINT8*  pbStationAddr   /* pointer to station address store */
)
{
UINT8 bCntr;

    /* Get station address, storred as 00 40 9E 00 xx yy, offset 0->5 */

    for (bCntr = 0; bCntr < 6; bCntr++)
        *(pbStationAddr++) = bEepromReadByte (20 + bCntr);

} /* vGetStationAddress () */


/*****************************************************************************
 * dCsrRead: read DWORD from the specified CSR
 *
 * RETURNS: the register contents
 */

UINT32 dCsrRead
(
    UINT16  wCsrOffset  /* destination register offset */
)
{
UINT32 dRegData;

    dRegData = inword (dDeviceIoBase + wCsrOffset);

    return (dRegData);

} /* dCsrRead () */
```


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

```
/*****************************************************************************
 * vCsrWrite: write DWORD to the specified CSR
 *
 * RETURNS: none
 */

void vCsrWrite
(
    UINT16  wCsrOffset, /* destination register offset */
    UINT32  dRegData    /* data to write */
)
{
    outword (dDeviceIoBase + wCsrOffset, dRegData);

} /* vCsrWrite () */


/*****************************************************************************
 * vCsrSet: set bits in the specified CSR
 *
 * RETURNS: none
 */

void vCsrSet
(
    UINT16  wCsrOffset, /* destination register offset */
    UINT32  dBitData    /* bit mask for bits to set (1 = set) */
)
{
UINT32 dRegData;

    dRegData = inword (dDeviceIoBase + wCsrOffset);
    outword (dDeviceIoBase + wCsrOffset, dRegData | dBitData);

} /* vCsrSet () */


/*****************************************************************************
 * vCsrClear: Clear bits in the specified CSR
 *
 * RETURNS: none
 */

void vCsrClear
(
    UINT16  wCsrOffset, /* destination register offset */
    UINT32  dBitData    /* bit mask for bits to clear (1 = clear) */
)
{
UINT32 dRegData;

    dRegData = inword (dDeviceIoBase + wCsrOffset);
    outword (dDeviceIoBase + wCsrOffset, dRegData & ~dBitData);

} /* vCsrClear () */
```


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

```
/*
 * Serial EEPROM programming utilities */
 */

void vEepromDelay (void)
{
UINT16 i, j;

    j = 0;

    for (i = 0; i < 10; i++)
        j++;

} /* vEepromDelay () */


void vEepromWriteBit
(
    UINT16 wDataBit
)
{
UINT32 dRegData;

    dRegData = (UINT32)(wDataBit & 0x0001) << 2;

    vCsrWrite (CSR9_ROM_CTL, CSR9__WRSEL | dRegData);
    vCsrWrite (CSR9_ROM_CTL, CSR9__WRSEL | CSR9__CK | dRegData);
    vCsrWrite (CSR9_ROM_CTL, CSR9__WRSEL | dRegData);

} /* vEepromWriteBit () */


UINT16 wEepromReadBit (void)
{
UINT32 dDataBit;

    vCsrWrite (CSR9_ROM_CTL, CSR9__RDSEL);
    vCsrWrite (CSR9_ROM_CTL, CSR9__RDSEL | CSR9__CK);
    vCsrWrite (CSR9_ROM_CTL, CSR9__RDSEL);

    dDataBit = (dCsrRead (CSR9_ROM_CTL) >> 3) & 0x00000001;

    return ((UINT16)dDataBit);

} /* wEepromReadBit () */


void vEepromClock (void)
{
    vCsrWrite (CSR9_ROM_CTL, CSR9__WRSEL);
    vCsrWrite (CSR9_ROM_CTL, CSR9__WRSEL | CSR9__CK);
    vCsrWrite (CSR9_ROM_CTL, CSR9__WRSEL);

} /* vEepromClock () */


UINT16 wEepromChipErase (void)
{
UINT16 wDeviceStatus;
UINT16 wTimer;

    wDeviceStatus = E_OK;

    vCsrWrite (CSR9_ROM_CTL, CSR9__SR | CSR9__CS); /* Enable interface */
    vEepromClock ();
```


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

```
    vEepromWriteBit (1); /* Message start bit */
    vEepromWriteBit (0);
    vEepromWriteBit (0); /* ERAL: op-code 00b, A5=1, A4=0, A3-A0=X */

    vEepromWriteBit (1); /* A5 */
    vEepromWriteBit (0); /* A4 */
    vEepromWriteBit (0); /* A3 */
    vEepromWriteBit (0); /* A2 */
    vEepromWriteBit (0); /* A1 */
    vEepromWriteBit (0); /* A0 */

    vCsrWrite (CSR9_ROM_CTL, CSR9__SR); /* CS low - begin erase */
    vEepromClock ();
    vEepromClock ();

    wTimer = 0;
    while ( (wTimer < 100) && (!wEepromReadBit ()) )
    {
        wTimer++;
        vGeneralDelay (1);
    }

    vCsrWrite (CSR9_ROM_CTL, CSR9__SR);
    vEepromClock ();
    vCsrWrite (CSR9_ROM_CTL, 0); /* Switch off interface */

    if (wTimer == 100)
        wDeviceStatus = E__EEPROM_ERASE;

    return (wDeviceStatus);

} /* wEepromChipErase () */


void vEepromWriteEnable (void)
{
    vCsrWrite (CSR9_ROM_CTL, CSR9__SR | CSR9__CS); /* Enable interface */
    vEepromClock ();

    vEepromWriteBit (1); /* Message start bit */
    vEepromWriteBit (0);
    vEepromWriteBit (0); /* EWEN: op-code 00b, A5=1, A4=1, A3-A0=X */

    vEepromWriteBit (1); /* A5 */
    vEepromWriteBit (1); /* A4 */
    vEepromWriteBit (0); /* A3 */
    vEepromWriteBit (0); /* A2 */
    vEepromWriteBit (0); /* A1 */
    vEepromWriteBit (0); /* A0 */

    vCsrWrite (CSR9_ROM_CTL, CSR9__SR); /* CS low */
    vEepromClock ();
    vEepromClock ();
    vCsrWrite (CSR9_ROM_CTL, 0);

} /* vEepromWriteEnable () */


void vEepromWriteDisable (void)
{
    vCsrWrite (CSR9_ROM_CTL, CSR9__SR | CSR9__CS); /* Enable interface */
    vEepromClock ();

    vEepromWriteBit (1); /* Message start bit */
```


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

```
    vEepromWriteBit (0);
    vEepromWriteBit (0); /* EWDS: op-code 00b, A5=0, A4=0, A3-A0=X */

    vEepromWriteBit (0); /* A5 */
    vEepromWriteBit (0); /* A4 */
    vEepromWriteBit (0); /* A3 */
    vEepromWriteBit (0); /* A2 */
    vEepromWriteBit (0); /* A1 */
    vEepromWriteBit (0); /* A0 */

    vCsrWrite (CSR9_ROM_CTL, CSR9__SR); /* CS low */
    vEepromClock ();
    vEepromClock ();
    vCsrWrite (CSR9_ROM_CTL, 0);

} /* vEepromWriteDisable () */


UINT16 wEepromWriteWord
(
    UINT16 wAddr,
    UINT16 wData
)
{
UINT16  wDeviceStatus;
UINT16  wTimer;
char    achBuffer[80];  /* text formatting buffer */
int     iIndex;

    wDeviceStatus = E_OK;

    vCsrWrite (CSR9_ROM_CTL, CSR9__SR | CSR9__CS); /* Enable interface */
    vEepromClock ();

    vEepromWriteBit (1); /* Message start bit */
    vEepromWriteBit (0);
    vEepromWriteBit (1); /* WRITE op-code 01b */

    for (iIndex=5; iIndex>=0; iIndex--)
        vEepromWriteBit (wAddr >> iIndex); /* addr */

    for (iIndex=15; iIndex>=0; iIndex--)
        vEepromWriteBit (wData >> iIndex); /* data */

    vCsrWrite (CSR9_ROM_CTL, CSR9__SR); /* CS low - begin write */
    vEepromClock ();
    vEepromClock ();

    wTimer = 0;
    while ( (wTimer < 100) && (!wEepromReadBit ()) )
    {
        wTimer++;
        vGeneralDelay (1);
    }

    vCsrWrite (CSR9_ROM_CTL, CSR9__SR);
    vEepromClock ();
    vCsrWrite (CSR9_ROM_CTL, 0); /* Switch off interface */

    if (wTimer == 100)
        wDeviceStatus = E__EEPROM_WRITE;

    return (wDeviceStatus);

} /* wEepromWriteWord () */
```


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

```
UINT8 bEepromReadByte
(
    UINT16 wAddr
)
{
UINT16  wWordAddr;
UINT16  wData;
int     iIndex;

    wWordAddr = (wAddr >> 1); /* convert byte address to word address */
    wData = 0;

    vCsrWrite (CSR9_ROM_CTL, CSR9__SR); /* Enable interface */

    vEepromClock ();

    vEepromWriteBit (1); /* Message start bit */
    vEepromWriteBit (1);
    vEepromWriteBit (0); /* READ op-code 10b */

    for (iIndex = 5; iIndex >= 0; iIndex--)
        vEepromWriteBit (wWordAddr >> iIndex); /* addr */

    for (iIndex = 15; iIndex >= 0; iIndex--)
        wData |= (wEepromReadBit () << iIndex); /* data */

    vCsrWrite (CSR9_ROM_CTL, 0); /* Switch off interface */

    /* Extract requested byte from word */

    if ((wAddr & 0x0001) != 0) /* if MSB requested, then shift */
        wData = wData >> 8;

    return ((UINT8)wData);

} /* bEepromReadByte () */


/*******************************************************************************
 * dComputeCrc: computes the CRC-32 of a buffer using a table based
 * approach [after Digital].
 *
 * RETURNS: none
 *
 */

UINT32 dComputeCrc
(
    UINT8* pbBuffer, /* pointer to data buffer to calculate CRC of */
    UINT16 wBufferSize /* size of this buffer */
)
{
UINT32  dThisCrc;   /* partial computation of CRC */
UINT16  wCounter;   /* buffer processing counter */
UINT8   bIndex;     /* index into the CRC table */

    dThisCrc = 0xFFFFFFFF ;

    for (wCounter = 0; wCounter < wBufferSize; wCounter++)
    {
        bIndex = (UINT8)(dThisCrc ^ (UINT32)(*pbBuffer++));
        dThisCrc = adCrcTable[bIndex] ^ (dThisCrc >> 8);
    }
```


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

```
    dThisCrc = ~dThisCrc;

    return (dThisCrc) ;

} /* vComputeCrc () */


/*****************************************************************************
 * vSelectPort: Set the chip up for the port and data rate to be used.
 *
 * RETURNS: none
 *
 */
void vSelectPort (UINT8 bPort, UINT8 bRate)
{
    switch (bPort)
    {
        case PT_SYM:

            /* Set SYM port as per the Digital data sheet */

            vCsrClear (CSR6_OP_MODE, CSR6__PS);
            vCsrWrite (CSR13_SIA_CON, 0);
            vCsrWrite (CSR14_SIA_TXRX, 0);
            vCsrSet (CSR6_OP_MODE, CSR6__PCS);
            vCsrSet (CSR6_OP_MODE, CSR6__SCR);

            if (bRate == SP_100)
            {
                vCsrClear (CSR6_OP_MODE, CSR6__TTM);
                vCsrSet (CSR6_OP_MODE, CSR6__PS);
            }
            else
                vCsrSet (CSR6_OP_MODE, CSR6__TTM);

            break;

        case PT_10BASET:

            /* Set 10 base T port */

            vCsrClear (CSR6_OP_MODE, CSR6__PS);
            vCsrWrite (CSR13_SIA_CON, 0);
            vCsrWrite (CSR14_SIA_TXRX, 0x7F3F);
            vCsrWrite (CSR13_SIA_CON, 1);

            break;

        default:
            break;
    }
}
```


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

4.4.1 Ethernet Data Definitions

The following listing contains data type and constant declarations used when programming the 21143 Ethernet controller.

```
/* defines */

 /* CSRs */

#define CSR0_BUS_MODE   0x00            /* bus mode */
#define     CSR0__WIE   0x01000000
#define     CSR0__RLE   0x00800000
#define     CSR0__RME   0x00200000
#define     CSR0__DBO   0x00100000
#define     CSR0__TAP   0x000E0000
#define     TAP_SHIFT   17
#define     CSR0__CAL   0x0000C000
#define     CAL_SHIFT   14
#define     CSR0__PBL   0x00003F00
#define     PBL_SHIFT   8
#define     CSR0__BLE   0x00000080
#define     CSR0__DSL   0x00000073
#define     DSL_SHIFT   2
#define     CSR0__BAR   0x00000002
#define     CSR0__SWR   0x00000001

#define CSR1_TX_POLL    0x08            /* transmit poll demand */

#define CSR2_RX_POLL    0x10            /* receive poll demand*/

#define CSR3_RX_BASE    0x18            /* receive list base */

#define CSR4_TX_BASE    0x20            /* transmit list base */

#define CSR5_STATUS 0x28                /* status */
#define     CSR5__LC    0x08000000
#define     CSR5__GPI   0x04000000
#define     CSR5__EB    0x03800000
#define     CSR5__TS    0x00700000
#define     TS_SHIFT    20
#define     TS_STOP     0
#define     TS_SUSPEND  6
#define     CSR5__RS    0x000E0000
#define     RS_SHIFT    17
#define     RS_STOP     0
#define     RS_WAIT_RX  3
#define     RS_SUSPEND  4
#define     CSR5__NIS   0x00010000
#define     CSR5__AIS   0x00008000
#define     CSR5__ERI   0x00004000
#define     CSR5__FBE   0x00002000
#define     CSR5__LNF   0x00001000
#define     CSR5__GTE   0x00000800
#define     CSR5__ETI   0x00000400
#define     CSR5__RWT   0x00000200
#define     CSR5__RPS   0x00000100
#define     CSR5__RU    0x00000080
#define     CSR5__RI    0x00000040
#define     CSR5__UNF   0x00000020
#define     CSR5__LNP_ANC 0x00000010
#define     CSR5__TJT   0x00000008
#define     CSR5__TU    0x00000004
#define     CSR5__TPS   0x00000002
#define     CSR5__TI    0x00000001
```


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

```
#define CSR6_OP_MODE    0x30            /* operation mode */
#define     CSR6__SC    0x80000000
#define     CSR6__RA    0x40000000
#define     CSR6__MBO   0x02000000
#define     CSR6__SCR   0x01000000
#define     CSR6__PCS   0x00800000
#define     CSR6__TTM   0x00400000
#define     CSR6__SF    0x00200000
#define     CSR6__HBD   0x00080000
#define     CSR6__PS    0x00040000
#define     CSR6__CA    0x00020000
#define     CSR6__TR    0x0000C000
#define     CSR6__ST    0x00002000
#define     CSR6__FC    0x00001000
#define     CSR6__OM    0x00000C00
#define     OM_SHIFT    10
#define     CSR6__FD    0x00000200
#define     CSR6__PM    0x00000080
#define     CSR6__PR    0x00000040
#define     CSR6__SB    0x00000020
#define     CSR6__IF    0x00000010
#define     CSR6__PB    0x00000008
#define     CSR6__HO    0x00000004
#define     CSR6__SR    0x00000002
#define     CSR6__HP    0x00000001

#define CSR7_INT_EN     0x38            /* interrupt enable */
#define     CSR7__LC    0x08000000
#define     CSR7__GPE   0x04000000
#define     CSR7__NIE   0x00010000
#define     CSR7__AIE   0x00008000
#define     CSR7__ERE   0x00004000
#define     CSR7__FBE   0x00002000
#define     CSR7__LFE   0x00001000
#define     CSR7__GTE   0x00000800
#define     CSR7__ETE   0x00000400
#define     CSR7__RWE   0x00000200
#define     CSR7__RSE   0x00000100
#define     CSR7__RUE   0x00000080
#define     CSR7__RIE   0x00000040
#define     CSR7__UNE   0x00000020
#define     CSR7__LPE_ANE 0x00000010
#define     CSR7__TJE   0x00000008
#define     CSR7__TUE   0x00000004
#define     CSR7__TSE   0x00000002
#define     CSR7__TIE   0x00000001

#define CSR8_MF_OC      0x40            /* missed frames/ overflow counter */
#define     CSR8__OCO   0x10000000
#define     CSR8__FOC   0x0FFE0000
#define     CSR8__MFO   0x00010000
#define     CSR8__MFC   0x0000FFFF

#define CSR9_ROM_CTL    0x48            /* boot ROM, serial ROM and MMII */
#define     CSR9__MDI   0x00080000
#define     CSR9__MII   0x00040000
#define     CSR9__MDO   0x00020000
#define     CSR9__MDC   0x00010000
#define     CSR9__RD    0x00004000
#define     CSR9__WR    0x00002000
#define     CSR9__BR    0x00001000
#define     CSR9__SR    0x00000800
#define     CSR9__REG   0x00000400
#define     CSR9__DATA  0x000000FF
```


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

```
#define     CSR9__DO    0x00000008
#define     CSR9__DI    0x00000004
#define     CSR9__CK    0x00000002
#define     CSR9__CS    0x00000001

#define     CSR9__RDSEL (CSR9__SR | CSR9__CS | CSR9__RD)
#define     CSR9__WRSEL (CSR9__SR | CSR9__CS | CSR9__WR)

#define CSR10_ROM_ADDR  0x50            /* boot ROM programming address */

#define CSR11_TIMER     0x58            /* general-purpose timer */
#define     CSR11__CON  0x00010000
#define     CSR11__TIMER 0x0000FFFF

#define CSR12_SIA_STAT  0x60            /* SIA status */
#define     CSR12__LPC  0xFFFF0000
#define     CSR12__LPN  0x00008000
#define     CSR12__ANS  0x00007000
#define     ANS_SHIFT   12
#define     CSR12__TRF  0x00000800
#define     CSR12__NSN  0x00000400
#define     CSR12__TRA  0x00000200
#define     CSR12__ARA  0x00000100
#define     CSR12__APS  0x00000008
#define     CSR12__LS10 0x00000004
#define     CSR12__LS100 0x00000002
#define     CSR12__MRA  0x00000001

#define CSR13_SIA_CON   0x68            /* SIA connectivity */
#define     CSR13__AUI  0x00000008
#define     CSR13__RST  0x00000001

#define CSR14_SIA_TXRX  0x70            /* SIA transmit and receive */
#define     CSR14__T4   0x00040000
#define     CSR14__TXF  0x00020000
#define     CSR14__TXH  0x00010000
#define     CSR14__TAS  0x00008000
#define     CSR14__SPP  0x00004000
#define     CSR14__APE  0x00002000
#define     CSR14__LTE  0x00001000
#define     CSR14__SQE  0x00000800
#define     CSR14__CLD  0x00000400
#define     CSR14__CSQ  0x00000200
#define     CSR14__RSQ  0x00000100
#define     CSR14__ANE  0x00000080
#define     CSR14__TH   0x00000040
#define     CSR14__CPEN 0x00000030
#define     CPEN_SHIFT  4
#define     CSR14__LSE  0x00000008
#define     CSR14__DREN 0x00000004
#define     CSR14__LBK  0x00000002
#define     CSR14__ECEN 0x00000001

#define CSR15_SIA_PORT  0x78            /* SIA and general-purpose port */
#define     CSR15__RMI  0x40000000
#define     CSR15__GI1  0x20000000
#define     CSR15__GI0  0x10000000
#define     CSR15__CWE  0x08000000
#define     CSR15__RME  0x04000000
#define     CSR15__GEI1 0x02000000
#define     CSR15__GEI0 0x01000000
#define     CSR15__LGS3 0x00800000
#define     CSR15__LGS2 0x00400000
#define     CSR15__LGS1 0x00200000
#define     CSR15__LGS0 0x00100000
```


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

```
#define     CSR15__MD   0x000F0000
#define     MD_SHIFT    16
#define     CSR15__HCKR 0x00008000
#define     CSR15__RMP  0x00004000
#define     CSR15__RWR  0x00000020
#define     CSR15__RWD  0x00000010
#define     CSR15__ABM  0x00000008
#define     CSR15__JCK  0x00000004
#define     CSR15__HUJ  0x00000002
#define     CSR15__JBD  0x00000001

 /* Receive and transmit descriptors */

#define RDES_0_STATUS   0           /* receive descriptor fields */
#define RDES_1_CTRL     4
#define RDES_2_BUFF1    8
#define RDES_3_BUFF2    12

#define RDES0__OWN  0x80000000      /* receive descriptor, status bits */
#define RDES0__FF   0x40000000
#define RDES0__FL   0x3FFF0000
#define RDES0__ES   0x00008000
#define RDES0__DE   0x00004000
#define RDES0__DT   0x00003000
#define RDES0__RF   0x00000800
#define RDES0__MF   0x00000400
#define RDES0__FS   0x00000200
#define RDES0__LS   0x00000100
#define RDES0__TL   0x00000080
#define RDES0__CS   0x00000040
#define RDES0__FT   0x00000020
#define RDES0__RW   0x00000010
#define RDES0__RE   0x00000008
#define RDES0__DB   0x00000004
#define RDES0__CE   0x00000002
#define RDES0__ZERO 0x00000001

#define RDES1__RER  0x02000000      /* receive descriptor, control bits */
#define RDES1__RCH  0x01000000
#define RDES1__RBS2 0x003FF800
#define RBS2_SHIFT  11
#define RDES1__RBS1 0x000007FF
#define TDES_0_STATUS   0           /* transmit descriptor fields */
#define TDES_1_CTRL     4
#define TDES_2_BUFF1    8
#define TDES_3_BUFF2    12

#define TDES0__OWN 0x80000000       /* transmit descriptor, status bits */
#define TDES0__ES   0x00008000
#define TDES0__TO   0x00004000
#define TDES0__LO   0x00000800
#define TDES0__NC   0x00000400
#define TDES0__LC   0x00000200
#define TDES0__EC   0x00000100
#define TDES0__HF   0x00000080
#define TDES0__CC   0x00000078
#define CC_SHIFT    3
#define TDES0__LF   0x00000004
#define TDES0__UF   0x00000002L
#define TDES0__DE   0x00000001

#define TDES1__IC   0x80000000      /* receive descriptor, control bits */
#define TDES1__LS   0x40000000
#define TDES1__FS   0x20000000
#define TDES1__FT1  0x10000000
```


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

```
#define TDES1__SET  0x08000000
#define TDES1__AC   0x04000000
#define TDES1__TER  0x02000000
#define TDES1__TCH  0x01000000
#define TDES1__DPD  0x00800000
#define TDES1__FT0  0x00400000
#define TDES1__TBS2 0x003FF800
#define TBS2_SHIFT  11
#define TDES1__TBS1 0x000007FF

typedef struct TAGTXDESC    /* transmit descriptor structure */
{
    UINT32 dStatus;     /* descriptor status */
    UINT32 dControl;    /* control bits and buffer sizes */
    UINT32 dBuffAddr;   /* absolute address of buffer */
    UINT32 dChain;      /* address of next descriptor in chain */

} tagTxDesc;

typedef struct TAGRXDESC    /* receive descriptor structure */
{
    UINT32 dStatus;     /* descriptor status */
    UINT32 dControl;    /* control bits and buffer sizes */
    UINT32 dBuffAddr;   /* absolute address of buffer */
    UINT32 dChain;      /* address of next descriptor in chain */

} tagRxDesc;

typedef struct TAGTXRING
{
    tagTxDesc* psTxDesc;   /* pointer to transmit descriptor */
    UINT8*     pbTxBuff;   /* pointer to descriptor's buffer */

} tagTxRing;

typedef struct TAGRXRING
{
    tagRxDesc* psRxDesc;   /* pointer to receive descriptor */
    UINT8*     pbRxBuff;   /* pointer to descriptor's buffer */

} tagRxRing;
```


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

4.5 SCSI Processor Interface Routines

This section contains a few subroutines and associated constants and data structures to provide a basic software-hardware interface to the 53C875 SCSI processor operating in narrow non-Ultra SCSI mode. Three routines are provided to initialize the processor chip, reset the SCSI bus and start the execution of a 53C875 SCRIPT. No code is provided for SCRIPTs themselves or for higher-level interfaces.

```
/*
 * Local error codes.
 */

#define E_OK                        0x0000
#define E_SELECTION_ERROR           0x0001
#define E_NOT_CMD_PHASE             0x0002
#define E_NOT_DATA_IN_PHASE         0x0003
#define E_NOT_DATA_OUT_PHASE        0x0004
#define E_NOT_MSG_IN_PHASE          0x0005
#define E_NOT_MSG_OUT_PHASE         0x0006
#define E_NOT_STATUS_PHASE          0x0007
#define E_SGE_UDC_BF_WTD            0x0008

#define E_UNEXPECTED_PHASE          0x0020
#define E_UNKNOWN_ERROR             0x0021
#define E_TIMEOUT_ERROR             0x0022

#define E_SENSE_BYTE_COUNT_ERROR    0x0030
#define E_SENSE_MAPPED_ERROR        0x0031
#define E_FILE_MARK                 0x0032
#define E_DEVICE_NOT_READY          0x0033
#define E_END_OF_MEDIUM             0x0034
#define E_MEDIUM_ERROR              0x0035
#define E_HW_ERROR                  0x0036
#define E_ILL_REQ_ERROR             0x0037
#define E_UNIT_ATTENTION            0x0038
#define E_DATA_PROT_ERROR           0x0040
#define E_END_OF_DATA               0x0041
#define E_UNCLASSIFIED              0x0042
#define E_ABORT_COMMAND_ERROR       0x0043
#define E_FILE_ID                   0x0044
#define E_IO_ERROR                  0x0045
#define E_NO_DEVICE_EXIST           0x0046
#define E_OMF_TYPE                  0x0047
#define E_OMF_OVERWRITE             0x0048
#define E_OMF_CHECKSUM              0x0049

#define NCR_OK                      0x0a
#define SCRIPT_OK                   0x00000004

/* Register addresses for 53C875 */

typedef struct ncr_810_struc
{
    UINT8 scntl0;       /* little endian*/
    UINT8 scntl1;
    UINT8 scntl2;
    UINT8 scntl3;
    UINT8 scid;
    UINT8 sxfer;
    UINT8 sdid;
    UINT8 gpreg;
    UINT8 sfbr;
    UINT8 socl;
    UINT8 ssid;
    UINT8 sbcl;
```


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

```
    UINT8 dstat;
    UINT8 sstat0;
    UINT8 sstat1;
    UINT8 sstat2;
    UINT32 dsa;
    UINT8 istat;
    UINT8 res1[3];
    UINT8 ctest0;
    UINT8 ctest1;
    UINT8 ctest2;
    UINT8 ctest3;
    UINT32 temp;
    UINT8 dfifo;
    UINT8 ctest4;
    UINT8 ctest5;
    UINT8 ctest6;
    UINT8 dbc[3];
    UINT8 dcmd;
    UINT32 dnad;
    UINT32 dsp;
    UINT32 dsps;
    UINT32 scratch_a;
    UINT8 dmode;
    UINT8 dien;
    UINT8 dwt;
    UINT8 dcntl;
    UINT32 adder;
    UINT8 sien0;
    UINT8 sien1;
    UINT8 sist0;
    UINT8 sist1;
    UINT8 slpar;
    UINT8 swide;
    UINT8 macntl;
    UINT8 gpcntl;
    UINT8 stime0;
    UINT8 stime1;
    UINT8 respid0;
    UINT8 respid1;
    UINT8 stest0;
    UINT8 stest1;
    UINT8 stest2;
    UINT8 stest3;
    UINT16 sidl;
    UINT16 res2;
    UINT16 sodl;
    UINT16 res3;
    UINT16 sbdl;
    UINT16 res4;
    UINT32 scratch_b;

} ncr_810_t;

/*
 * SCSI control structure - this structure and the fields within it MUST be
 * aligned on 4-byte boundaries.
 */

typedef struct ncr_struc
{
    UINT8 dummy;            /* always 0 */
    UINT8 period_offset;    /* synchronuos offset and period */
    UINT8 target_id;        /* scsi target id */
    UINT8 config;           /* always 0 for 710, scntl3 for 810*/
    UINT32 byte_count;      /* number of bytes to transfer */
```


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

```
    UINT32 buffer_address;  /* physical buffer address */
    UINT32 status_count;    /* byte count of status */
    UINT32 status;          /* transfer status address */
    UINT32 identify_count;  /* byte count identify message */
    UINT32 identify_msg;    /* identify message buffer address */
    UINT32 cmnd_count;      /* byte count cmnd buffer */
    UINT32 cmnd_buffer;     /* command buffer address */
    UINT32 msg_in_count;    /* byte count message in */
    UINT32 msg_in_buffer;   /* message in buffer address */
    UINT32 sense_count;     /* request sense buffer byte count*/
    UINT32 sense_buffer;    /* request sense buffer address */
};


/* External subroutines */

/* General-purpose routines */

extern void delay (UINT32 milliseconds);
extern UINT32 get_address (void *ptr);

/* Forward references */

UINT32 start_scsi ();
UINT32 init_scsi ();
UINT32 reset_scsi ();
UINT32 scsi_get_address (void *ptr);

/* Local data structures */

/*
 * The 'ncr_struc' structure below, and each element within it, MUST be
 * positioned on a 4-byte boundary. The structure is used by the SCRIPTS
 * processor during SCSI operations.
 */

static struct ncr_struc ncr;
static struct ncr *ncr_p;

#pragma ALIGN (ncr)


/*
 * The 'ncr_810_t' structure used in the remainder of this code MUST be
 * positioned at the correct memory address. This structure contains the
 * memory-mapped register set of the 53C875 chip.
 * To position this structure correctly, indirect references are used to
 * access the device, using the 'ncr_810_ptr' pointer, which is set up
 * during initialization to point at a suitably-positioned variable
 * in an alternative data segment. Hence, in this case, 'ncr_reg_base' is
 * declared 'far'. The segment containing 'ncr_reg_base' would be correctly
 * located by the BLD386 utility.
 * Other methods of positioning the 'ncr_810_t' structure are possible,
 * but are not described here.
 */

extern int far ncr_reg_base;
far ncr_810_t *ncr_810_ptr;
```


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

```
/*********************************************************************
 *
 * Title: init_scsi()
 *
 *  Abstract: This procedure initializes all registers which will be
 *   needed for script executing.
 *      Since this sets up globals for chip registers,
 *     it must be called first
 *
 *********************************************************************/

UINT32 init_scsi(void)
{
UINT32 delay;
UINT32 exception;
static UINT32 sense_addr;
UINT8 reset_status;

    ncr_810_ptr = (far ncr_810_t *) &ncr_reg_base;

    /* set pointer to register structure */
    /* disable Controller's ready */

    ncr_810_ptr->dcntl      = 0x00;     /* normal arbitration */
    ncr_810_ptr->scntl3     = 0x33;     /* set clock to 37-50 MHz */
    ncr_810_ptr->istat      = 0x40;     /* reset SCSI Controller */
    delay(10);                          /* wait a little bit */
    ncr_810_ptr->istat      = 0;        /* clear this reset condition */
    ncr_810_ptr->ctest3     = 0x0c;     /* flush the fifos */
    delay(10);                          /* wait a little bit */
    ncr_810_ptr->ctest3     = 0;
    ncr_810_ptr->scntl0     = 0xc4;     /* full arbitration, select with ATN,
                                           parity generation, initator mode */
    ncr_810_ptr->sdid       = 0x00;     /* selection id = 0 */
    ncr_810_ptr->sien0      = 0x89;     /* M/A, SGE, PAR */
    ncr_810_ptr->sien1      = 0x04;     /* STO */
    ncr_810_ptr->dien       = 0x26;     /* SIR,WTD,BF DMA interrupts */
    ncr_810_ptr->scid       = get_scsi_id();
    ncr_810_ptr->sxfer      = 0;        /* asynchronous mode only ! */
    ncr_810_ptr->ctest4     = 0;        /* no MUX mode */
    ncr_810_ptr->ctest5     = 0;        /* no reset,no request,no ack */
    ncr_810_ptr->ctest0     = 0;
    ncr_810_ptr->dmode      = 0x89;     /* Burst len=8,manual,read line enable*/
    ncr_810_ptr->dwt        = 52;       /* DMA watchdog timeout = 50 us */
    ncr_810_ptr->stime0     = 0x0c;     /* Selection timeout of 205 ms */
    ncr_810_ptr->dbc[0]     = 0;        /* DMA byte count = 0 */
    ncr_810_ptr->dbc[1]     = 0;
    ncr_810_ptr->dbc[2]     = 0;
    ncr_810_ptr->dsps       = 0;        /* reset DSPS - registers */

    ncr_p = &ncr;       /* init data structure and pointers */

    ncr_p->dummy            = 0;
    ncr_p->config           = 0x33;
    ncr_p->period_offset    = 0;
    ncr_p->target_id        = 0;
    ncr_p->byte_count       = 0;
    ncr_p->buffer_address   = 0;

    /* give him the base of data structure */

    ncr_810_ptr->dsa = scsi_get_address(&ncr);

    /*
```


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

```
     * Now reset the SCSI.
     */

    exception = reset_scsi();

    return (exception);
}


/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *
 *  TITLE: reset_scsi
 *
 *  ABSTRACT: This procedure resets the SCSI Bus. A status code is
 *      returned which is 0 for success, or a non-zero value
 *     indicating the failure mode.
 *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

UINT32 reset_scsi (void)
{
    UINT16 count;
    UINT8 scntl;
    UINT8 istatus;
    UINT8 dummy;

    count = 0;
    istatus = ncr_810_ptr->istat;

    ncr_810_ptr->istat = (istatus | 0x80);  /* abort operation */
    istatus = 0x00;

    while (((istatus & 0x03) != 0x01) && (count < 0x2000))
    {
        istatus = ncr_810_ptr->istat;
        scntl = ncr_810_ptr->sist0 & 0x02; /*remove any pending intrpts*/
        dummy = ncr_810_ptr->sist1; /*WARNING:rd sists as pair to clr istat*/

        count++;
    }

    if (count >= 0x2000)
    {
        return (0x90);
    }

    ncr_810_ptr->istat = 0x0;
    scntl = ncr_810_ptr->dstat & 0x10;

    if (scntl == 0)
    {
        return (0x91);
    }

    scntl = ncr_810_ptr->scntl1;
    ncr_810_ptr->scntl1 = (scntl | 0x08);  /* set the RST bit */
    delay(2);

    scntl = scntl & 0xf7;
    ncr_810_ptr->scntl1 = scntl;      /* reset the RST bit */

    scntl = 0x0;
    istatus = 0x0;
    count = 0x0;
```


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

```
    while ((scntl == 0) && (istatus == 0) && (count < 0x2000))
    {
        istatus = ncr_810_ptr->istat & 0x02; /* check if reset int occurred */
        scntl = ncr_810_ptr->sist0 & 0x02; /* check if RST/ active */

        dummy = ncr_810_ptr->sist1;
        count++;
    }

    if (count >= 0x2000)
    {
        return (0x92);
    }

    delay(10);
    scntl = ncr_810_ptr->sist0 & 0x02;  /* check if RST/ bit is cleared */

    dummy = ncr_810_ptr->sist1;

    if ((scntl & 0x02) != 0x0)
    {
        return (0x93);
    }

    delay(1);

    scntl = ncr_810_ptr->istat & 0x02;  /* check if int bit is cleared */

    if (scntl != 0x0)
    {
        return (0x94);
    }

    scntl = ncr_810_ptr->scntl1;
    ncr_810_ptr->scntl1 = (scntl & 0x0ef);   /* force disconnect */

    delay(1000);           /* 1 second delay */

    return (0x00);
}


/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
 *
 *  TITLE: start_scsi
 *
 *  ABSTRACT: This procedure starts the SCSI phase sequences.
 *
 *   The procedure returns a status code representing a hardware problem.
 *
 * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

UINT32 start_scsi
(
    UINT32 *script_ptr,
    UINT8 *buffer_ptr,
    UINT32 num_bytes,
    UINT8 target_id,
    UINT32 *actual_ptr
)
{
    UINT32 done;   /* Flag for ending DO WHILE LOOP */
    UINT8 byte_reg;  /* Used to read a byte register */
    UINT8 sstat_reg;  /* Used to read the interrupt status*/
    UINT8 dsps_reg;   /* Used to read the interrupt status*/
```


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

```
    UINT32 counter;
    UINT32 status_reg; /* status registers */
    UINT8 ss[2];   /* scsi status registers */
    UINT32 sto;    /* select timeout flag */
    UINT32 exception;  /* function result */
    UINT32 bytes_left; /* Whats left after the DATA_IN phase */
    UINT32 *pp;
    UINT32 dbc_reg;
    UINT32 dsp_reg;
    UINT8 phase_mis;


    /* Clean first */

    exception = 0;
    phase_mis = 0;

    /* setup data buffer addresses */

    ncr_p->buffer_address = scsi_get_address(buffer_ptr);

    /* setup byte count */

    ncr_p->byte_count = num_bytes;

    done = FALSE;

    ncr_810_ptr->dsa = scsi_get_address(&ncr);
    ncr_810_ptr->dbc[0] = 0;
    ncr_810_ptr->dbc[1] = 0;
    ncr_810_ptr->dbc[2] = 0;

    ncr_p->target_id = target_id; /* setup target id */

    dbc_reg = 0;
    dsp_reg = 0;

    ncr_810_ptr->dsp = scsi_get_address(script_ptr);
    ncr_810_ptr->dcntl = ncr_810_ptr->dcntl | 0x04;

    /**** Now wait for the response, with a counter for problems.****/

wait_for_int:
    counter = 0;

another_int:
    byte_reg = 0;

    while (byte_reg == 0 && counter < 0xffffffL)
    {
        /* any interrupts pending ? */
        byte_reg = ncr_810_ptr->istat & 0x03;
        counter = counter + 1;
    }

    if (counter >= 0xffffffL)
    {
        /*****************************************************************
         * Timeout waiting for an interrupt - take appropriate action for
         * the operating environment
         *****************************************************************/
    }

    /*
     * The normal/correct situation at this point is a 'Command Complete'
```


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

```
 * from the Script Processor, Check this condition first
 * and act accordingly.
 *
 * Any other possibility is an error condition.
 */

pp = (UINT32 *) &ncr_810_ptr->dstat;

ss[0] = ncr_810_ptr->sist0; /*need to read scsi status to remove int*/
ss[1] = ncr_810_ptr->sist1;

sstat_reg = ss[0];

if (sstat_reg & 0x04)
{
    /* fatal error (expected) so restart scripts and try again */

    ncr_810_ptr->dcntl = ncr_810_ptr->dcntl | 0x04;
    goto another_int;
}

status_reg = *pp; /* read status registers */

pp = (UINT32 *) &ncr_810_ptr->dbc[0];

if (!phase_mis)
{
    /* save the state of dbc (important for phase mismatch conditions) */

    dbc_reg = *pp & 0x00ffffff;
}

pp = (UINT32 *) &ncr_810_ptr->dsp;

dsp_reg = *pp; /* save script pointer */

if ((byte_reg & 0x01) && (status_reg & SCRIPT_OK))
{
    /* script interrupt instruction received ? */
    dsps_reg = ncr_810_ptr->dsps;

    /* check for problems */
    if (dsps_reg != NCR_OK)   /* end of script interrupt ? */
    {
        switch (dsps_reg)
        {
            /* selection failed */

            case 0x80:
                exception = E_SELECTION_ERROR;
                goto exit;

            /* error_not_cmd_phase */

            case 0x01:
                exception = E_NOT_CMD_PHASE;
                break;

            /* error_not_data_in_phase */

            case 0x02:
                exception = E_NOT_DATA_IN_PHASE;
                break;

            /* error_not_data_out_phase */
```


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

```
                case 0x03:
                    exception = E_NOT_DATA_OUT_PHASE;
                    break;

                /* error_not_msg_in_phase */

                case 0x04:
                    exception = E_NOT_MSG_IN_PHASE;
                    break;

                /* error_not_msg_out_phase */

                case 0x05:
                    exception = E_NOT_MSG_OUT_PHASE;
                    break;

                /* error_not_status_phase */

                case 0x06:
                    exception = E_NOT_STATUS_PHASE;
                    break;

                /* error_unexpected_phase */

                case 0x07:
                    exception = E_UNEXPECTED_PHASE;
                    break;

                /* unknown error condition */

                default:
                    exception = E_UNKNOWN_ERROR;
                    break;
            }

            goto exit;

        } /* end of script interrupt */

        else
        {
            /* dsps == NCR_OK ! */
            exception = E_OK;
        }
    }

    else if ((status_reg & 0x22) || (ss[0] & 0x1c) || (ss[1] & 0x04))
    {
        /* check for the other problems */

        if (sstat_reg & 0x80) /*phase mismatch*/
        {
            phase_mis++;

            /* we had a phase mismatch during transfer, try to
               get the status, restart the 810 */

            /* now restart at dsp */

            ncr_810_ptr->dcntl = ncr_810_ptr->dcntl | 0x04;

            goto wait_for_int;
        }
```


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

```
            sto = (ss[1] & 0x04) ? TRUE : FALSE;

            if (sto == TRUE)
            {
                /* scsi bus timeout or selection timeout (STO) */

                exception = E_TIMEOUT_ERROR;
                goto exit;
            }

            else if ((sstat_reg & 0x08) || (status_reg & 0x00000022))
            {
                /* scsi gross error (SGO) or */
                /* unexpected disconnect (UDC) or */
                /* BF or WTD */

                exception = E_SGE_UDC_BF_WTD;
                goto exit;
            }
        }   /* other problems ? */

        /* this point is reached only if script has returned status ok */

        /*
         * Get the number of bytes left to transfer,
         * normally this would be zero.
         */

        if (phase_mis)
            bytes_left = (UINT32) dbc_reg;
        else
            bytes_left = 0;

        /*
         * Set the Actual.
         */

        *actual_ptr = num_bytes - bytes_left;

    exit:
        return (exception);

    }


    /* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
     *
     *   TITLE: scsi_get_address
     *
     * ABSTRACT:    This procedure computer the physical
     *      address of data to be accessed by the scsi processor.
     * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

    UINT32 scsi_get_address(void *addr)
    {
        UINT32 tmpaddr;

        tmpaddr = get_address((UINT8 *)addr);
        return (tmpaddr);
    }
```


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

4.6 VME to PCI Configuration

The code fragments below demonstrate one possible initialization and configuration sequence for the Tundra Universe chip, which provides the VME to PCI bus interface on this board.

```
#define IMAGE_EN    0x80000000

/* PCI Slave Image Control */

#define LSI_D32     0x00800000
#define LSI_A32     0x00020000

#define LSI0_CTL    0x100
#define LSI0_BS     0x104
#define LSI0_BD     0x108
#define LSI0_TO     0x10c

#define LSI1_CTL    0x114
#define LSI1_BS     0x118
#define LSI1_BD     0x11c
#define LSI1_TO     0x120

#define LSI2_CTL    0x128
#define LSI2_BS     0x12c
#define LSI2_BD     0x130
#define LSI2_TO     0x134

#define LSI3_CTL    0x13c
#define LSI3_BS     0x140
#define LSI3_BD     0x144
#define LSI3_TO     0x148

#define SCYC_CTL    0x170
#define SCYC_ADDR   0x174
#define SCYC_EN     0x178
#define SCYC_CMP    0x17c
#define SCYC_SWP    0x180

#define LMISC       0x184
#define SLSI        0x188
#define L_CMDERR    0x18c
#define LAERR       0x190

#define DCTL        0x200
#define DTBC        0x204
#define DLA         0x208
#define DVA         0x210
#define DCPP        0x218
#define DGCS        0x220
#define D_LLUE      0x224

#define LINT_EN     0x304
#define VSW_IACK    0x1000
#define LINT_MAP0   0x308
#define LINT_MAP1   0x30c
#define VINT_EN     0x310
#define SW_IACK_EN  0x1000

#define VINT_STAT   0x314
#define ACKLINT0    0x01

#define VINT_MAP0   0x318
#define VINT_MAP1   0x31c
```


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

```
#define SW_IACKMASK 0x70000

#define STATID      0x320
#define V1_STATID   0x324
#define V2_STATID   0x328
#define V3_STATID   0x32c
#define V4_STATID   0x330
#define V5_STATID   0x334
#define V6_STATID   0x338
#define V7_STATID   0x33c

#define MAST_CTL    0x400
#define VRL2        0x00800000
#define VRL3        0x00c00000
#define VRLMASK     0x00c00000

#define MISC_CTL    0x404
#define VARBTO      0x03000000

#define MISC_STAT   0x408
#define USER_AM     0x40c

#define VSI_DATA    0x00400000
#define VSI_NOPRIV  0x00100000
#define VSI_A32     0x00020000

#define VSI0_CTL    0xf00
#define VSI0_BS     0xf04
#define VSI0_BD     0xf08
#define VSI0_TO     0xf0c

#define VSI1_CTL    0xf14
#define VSI1_BS     0xf18
#define VSI1_BD     0xf1c
#define VSI1_TO     0xf20

#define VSI2_CTL    0xf28
#define VSI2_BS     0xf2c
#define VSI2_BD     0xf30
#define VSI2_TO     0xf34

#define VSI3_CTL    0xf3c
#define VSI3_BS     0xf40
#define VSI3_BD     0xf44
#define VSI3_TO     0xf48

#define VRAI_CTL    0xf70
#define VRAI_BS     0xf74

#define VCSR_CTL    0xf80
#define VCSR_TO     0xf84
#define V_AMERR     0xf88
#define VAERR       0xf8c

#define VCSR_CLR    0xff4
#define VCSR_SET    0xff8
#define VCSR_BS     0xffc
```


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

```
/* externals */

extern Int bootLine;
extern char * sysMemTopPhys;
extern struct
{
    unsigned long physicalAddr;
    unsigned long len;
} sysVMEMemory;


/* globals */

unsigned long universeBase;
unsigned char universeInt;

/* statics */

static int firstSwInt = TRUE;


/*****************************************************************************
 * universeInit - initialize the Universe VME/PCI bridge chip
 *
 * Set up the PCI/VME slave images and any interrupts as necessary*
 *
 * RETURNS: OK or ERROR if chip not found.
 */

unsigned long universeInit
(
    int intType,
    int intLevel,
    int *pIrq
)
{
BOOT_PARAMS params;
int i;
unsigned long   val;
unsigned char*  reg;
unsigned char   deviceId;
unsigned char   isThere;
unsigned short  pciCommand;

    deviceId = 0;

    isThere = PciBusScan (PCI_BUS_0, NEWBRIDGE_VENDOR_ID, UNIVERSE_MODULE_ID,
                          &deviceId);

    if (isThere)
    {
        /* Read command register */

        PciRead (PCI_BUS_0, deviceId, PCI_FUNC_0, PCI_COMMAND,
                 (unsigned char *)&pciCommand, (unsigned char)2);

        /* Enable Memory and PCI Bus master access */

        pciCommand |= (PCI_ENABLE_MEMORY | PCI_ENABLE_MASTERING |
                       PCI_ENABLE_SERR);

        PciWrite (PCI_BUS_0, deviceId, PCI_FUNC_0, PCI_COMMAND,
                 (unsigned char *)&pciCommand, (unsigned char)2);

        /* Now read the base address */
```


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

```
        PciRead (PCI_BUS_0, deviceId, PCI_FUNC_0, PCI_BASE_ADDR_0,
                    (unsigned char *)&universeBase, (unsigned char)4);

        /* Exclude bottom 2 bits of Memory address */

        universeBase &= 0xfffffffc;

        /* And finally... the interrupt level (IRQ) */

        PciRead (PCI_BUS_0, deviceId, PCI_FUNC_0, PCI_INTERRUPT_LINE,
                    (unsigned char *)&universeInt, (unsigned char)1);

        *pIrq = universeInt;
    }

    else
    {
        errno = ENODEV;
        return (ERROR);
    }

    /* Disable all PCI slave images */

    reg = (unsigned char *)(universeBase + LSI0_CTL);

    for (i = 0; i < 4; i++)
    {
        val = *(unsigned long *)reg;
        val &= ~IMAGE_EN;
        *(unsigned long *)reg = val;
        reg += 0x14;
    }

    /* Disable all VME slave images */

    reg = (unsigned char *)(universeBase + VSI0_CTL);

    for (i = 0; i < 4; i++)
    {
        val = *(unsigned long *)reg;
        val &= ~IMAGE_EN;
        *(unsigned long *)reg = val;
        reg += 0x14;
    }

    /* Set VME bus arbitration timeout to infinity */

    reg = (unsigned char *)(universeBase + MISC_CTL);
    val = *(unsigned long *)reg;
    val &= (~VARBTO);
    *(unsigned long *)reg = val;

    /* Set VME bus request level to 3 */

    reg = (unsigned char *)(universeBase + MAST_CTL);
    val = *(unsigned long *)reg;
    val &= (~VRLMASK);              /* clear the VRL bits */
    val |= VRL3;                    /* set to level 3 */
    *(unsigned long *)reg = val;

/* Prior to executing this piece of code the MMU must have been
 * initialized and the virtual and physical addresses modified for
 * the off board case.
 */
```


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

```
    bootStringToStruct ((char*)bootLine,&params);

    if (params.procNum)
    {
        /* Set up PCI slave image to access offboard memory.
         * Offboard shared memory is at VME address 0 and at a PCI
         * address at the top of DRAM.
         */

        *(unsigned long *)(universeBase + LSI0_BS) =
            (unsigned long)sysVMEMemory.physicalAddr;
        *(unsigned long *)(universeBase + LSI0_BD) =
            (unsigned long)sysVMEMemory.physicalAddr +
            (unsigned long)sysVMEMemory.len;

        /* Allow for VME addresses starting at 0 */

        *(unsigned long *)(universeBase + LSI0_TO) =
            (~(unsigned long)sysVMEMemory.physicalAddr) + 1;

        /* Enable image with 32 bit VME address and data */

        *(unsigned long *)(universeBase + LSI0_CTL) = IMAGE_EN | LSI_D32 |
                                                      LSI_A32;
    }

    else
    {
        /* Set up VME slave image so that other boards can access onboard
         * memory. Map VME address 0 onto PCI address 1Mb below the top of
         * DRAM which is the shared memory location.
         */

        *(unsigned long *)(universeBase + VSI0_BS) = 0;
        *(unsigned long *)(universeBase + VSI0_BD) =
            (unsigned long)sysVMEMemory.len;
        *(unsigned long *)(universeBase + VSI0_TO) =
            (unsigned long)sysVMEMemory.physicalAddr;

        /* Enable image with 32 bit VME address and data */

        *(unsigned long *)(universeBase + VSI0_CTL) = IMAGE_EN | VSI_A32 |
                                                      VSI_DATA | VSI_NOPRIV;
    }

    /* Map all unused interrupts onto an interrupt other than LINT 0 i.e 7 */

    *(unsigned long *)(universeBase + LINT_EN) &= 0xffff0800;
    *(unsigned long *)(universeBase + LINT_MAP0) |= 0x77777777;
    *(unsigned long *)(universeBase + LINT_MAP1) |= 0x77770777;

    /* Mask out all interrupts by mapping them onto VIRQ0 */

    *(unsigned long *)(universeBase + VINT_EN) &= 0xffffe000;
    *(unsigned long *)(universeBase + VINT_MAP0) &= 0x88888888;
    *(unsigned long *)(universeBase + VINT_MAP1) &= 0xfff8f888;

    return (OK);
}
```


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

```
/*******************************************************************************
 * sysLocalToBusAdrs - convert local address to bus address
 *
 * Given a local memory address, this routine returns the VME address
 * that would have to be accessed to get to that byte.
 *
 * RETURNS: OK, or ERROR if unable to get to that local address from the bus
 */

unsigned long sysLocalToBusAdrs
(
    int     adrsSpace,  /* bus address space in which busAdrs resides, */
                        /* use address modifier codes as defined in */
                        /* vme.h, such as VME_AM_STD_SUP_DATA */
    char*   localAdrs,  /* local address to convert */
    char**  pBusAdrs    /* where to return bus address */
)
{
    if ((localAdrs < 0) || (localAdrs >= (char *) sysMemTopPhys))
    {
        /* this is off-board memory - just return local address */

        *pBusAdrs = localAdrs;
    }

    else
    {
        /* This is on-board memory - map to bus address space - the
         * memory is placed in STD space at address LOCAL_MEM_BUS_ADRS.
         */

        if (((adrsSpace != VME_AM_STD_SUP_PGM) &&
            (adrsSpace != VME_AM_STD_SUP_DATA) &&
            (adrsSpace != VME_AM_STD_USR_PGM) &&
            (adrsSpace != VME_AM_STD_USR_DATA)))
            return (ERROR);

        *pBusAdrs = (char*) ((int) localAdrs - (int)sysVMEMemory.physicalAddr);
    }

    return (OK);
}


/*******************************************************************************
 * sysBusToLocalAdrs - convert bus address to local address
 *
 * Given a VME memory address, this routine returns the local address
 * that would have to be accessed to get to that byte.
 *
 * RETURNS: OK, or ERROR if unknown address space
 */

unsigned long sysBusToLocalAdrs
(
    int     adrsSpace,  /* bus address space in which busAdrs resides, */
                        /* use address modifier codes as defined in */
                        /* vme.h, such as VME_AM_STD_SUP_DATA */
    char*   busAdrs,    /* bus address to convert */
    char**  pLocalAdrs  /* where to return local address */
)
{
    switch (adrsSpace)
    {
        case VME_AM_SUP_SHORT_IO:
```


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

```
        case VME_AM_USR_SHORT_IO:

            *pLocalAdrs = (char *) (0xffff0000 | (int) busAdrs);
            return (OK);

        case VME_AM_STD_SUP_ASCENDING:
        case VME_AM_STD_SUP_PGM:
        case VME_AM_STD_SUP_DATA:
        case VME_AM_STD_USR_ASCENDING:
        case VME_AM_STD_USR_PGM:
        case VME_AM_STD_USR_DATA:
        case VME_AM_EXT_SUP_ASCENDING:
        case VME_AM_EXT_SUP_PGM:
        case VME_AM_EXT_SUP_DATA:
        case VME_AM_EXT_USR_ASCENDING:
        case VME_AM_EXT_USR_PGM:
        case VME_AM_EXT_USR_DATA:

            if (busAdrs < (char *) sysMemTopPhys)
            {
                /* this is on-board memory - just return bus address */

                *pLocalAdrs = busAdrs;
            }

            else
            {
                *pLocalAdrs = (char *)((int) busAdrs +
                                       (int) sysVMEMemory.physicalAddr);
            }

            return (OK);

        default:

            return (ERROR);
    }
}


/*******************************************************************************
 * sysIntDisable - disable interrupt level
 *
 * This routine disables the specified interrupt level.
 *
 * RETURNS: OK or ERROR if intLevel not in range 1 - 7
 */

unsigned long sysIntDisable
(
    int intLevel    /* interrupt level to disable */
)
{
    if ((intLevel < 1) || (intLevel > 7))
        return (ERROR);

    /* Disable the relevant VME interrupt level */

    *(unsigned long *)(universeBase + LINT_EN) &= ~(1 << intLevel);


    return (OK);
}
```


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

```
/*******************************************************************************
 * sysIntEnable - enable interrupt level
 *
 * This routine enables the specified VME interrupt level.
 *
 * RETURNS: OK or ERROR if intLevel not in range 1 - 7
 */

unsigned long sysIntEnable
(
    int intLevel    /* interrupt level to enable */
)
{
    if ((intLevel < 1) || (intLevel > 7))
        return (ERROR);

    /* Map required level onto LINT0 */

    *(unsigned long *)(universeBase + LINT_MAP0) &=
        (~(0x07 << (4 * intLevel)));

    /* Now enable the releveant VME interrupt */

    *(unsigned long *)(universeBase + LINT_EN) |= (1 << intLevel);

    return (OK);
}


/*******************************************************************************
 * sysBusIntAck - acknowledge VME bus interrupt
 *
 * This routine acknowledges the specified interrupt.
 *
 * RETURNS: NULL
 */

int sysBusIntAck
(
    int intLevel /* interrupt level to acknowledge */
)
{
    if (intLevel > 0 && intLevel < 8)
    {
        /* Clear the local interrupt */

        *(unsigned long *)(universeBase + LINT_STAT) = (1 << intLevel);
    }

    return (NULL);
}


/*******************************************************************************
 * sysBusIntGen - generate interrupt
 *
 * This routine generates a VME bus interrupt.
 *
 * RETURNS: OK or ERROR if level not in range 1 - 7
 */

unsigned long sysBusIntGen
(
    int level,
    int vector
```


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

```
)
{
unsigned long *pIntStatus;

    if ((level < 1) || (level > 7))
    {
        return (ERROR);
    }

    else
    {
        /* Map the VME interrupt level onto the sw iack interrupt */

        *(unsigned long *)(universeBase + VINT_MAP1) &= ~SW_IACKMASK;
        *(unsigned long *)(universeBase + VINT_MAP1) |= (level << 16);

        /* Next wait for the previous interrupt to be acknowledged
         * only if there was one.
         */

        if (firstSwInt)
            firstSwInt = FALSE;
        else
        {
            pIntStatus = (unsigned long *)(universeBase + LINT_STAT);

            while ((*pIntStatus & VSW_IACK) == 0)
            {
            }
        }

        /* Toggle the interrupt bit low then high */

        *(unsigned long *)(universeBase + VINT_EN) &= ~SW_IACK_EN;
        *(unsigned long *)(universeBase + VINT_EN) |= SW_IACK_EN;

        return (OK);
    }
}
```


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

4.7 Warm Reset via PORT-92 or Keyboard Controller

The board can be reset through either the keyboard controller or port-92. The port-92 method generally produces a faster reset.

```
/*
 * Define one of these to select the reset type
 */

/* #define RESET_MODE_PORT92 */
#define RESET_MODE_KEYBOARD

/* Legacy I/O ports */

#define PORT92          0x92    /* legacy port-92 */
#define KBD_COMMAND     0x64    /* legacy keyboard command port (write) */
#define KBD_STATUS      0x64    /* legacy keyboard status port (read) */

#define KBDSTATUS_INPUT 0x02    /* input buffer status bit */
#define KBDCMD_RESET    0xFE    /* 'reset' command (bit-0 low) */

void vWarmReset (void)
{
    UINT8   bTemp;      /* temporary BYTE */

#ifdef RESET_MODE_PORT92

    /* Do a warm reset via port-92 */

    bTemp = inbyte (PORT92);
    outbyte (PORT92, bTemp | 0x01);     /* pulses reset */

#endif

#ifdef RESET_MODE_KEYBOARD

    /* Check keyboard status port to see if OK to write */

    do {
        bTemp = inbyte (KBD_STATUS);

    } while ((bTemp & KBDSTATUS_INPUT) != 0);

    outbyte (KBD_COMMAND, KBDCMD_RESET);    /* reset the system */

    /* Wait for the command to be executed (may not reach this point
     * if reset propagates sufficiently fast)
     */

    do {
        bTemp = inbyte (KBD_STATUS);

    } while ((bTemp & KBD_STATUS_INPUT) != 0);

    for (;;);       /* endless loop to allow reset to propagate */

#endif
```


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

This page has been left intentionally blank.

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

Functional Description

5.1 General

In order to fully understand the operations of the VP PSE/P3x board, the user may wish to understand some of the details of the board hardware. This chapter offers a detailed description for each of the functional areas of the board. For further guidance on software issues refer to Chapter 4, Programming.

5.2 Functional Description

Figure 5-1 shows a functional overview of the VP PSE/P3x. The following sections give details of each functional block.

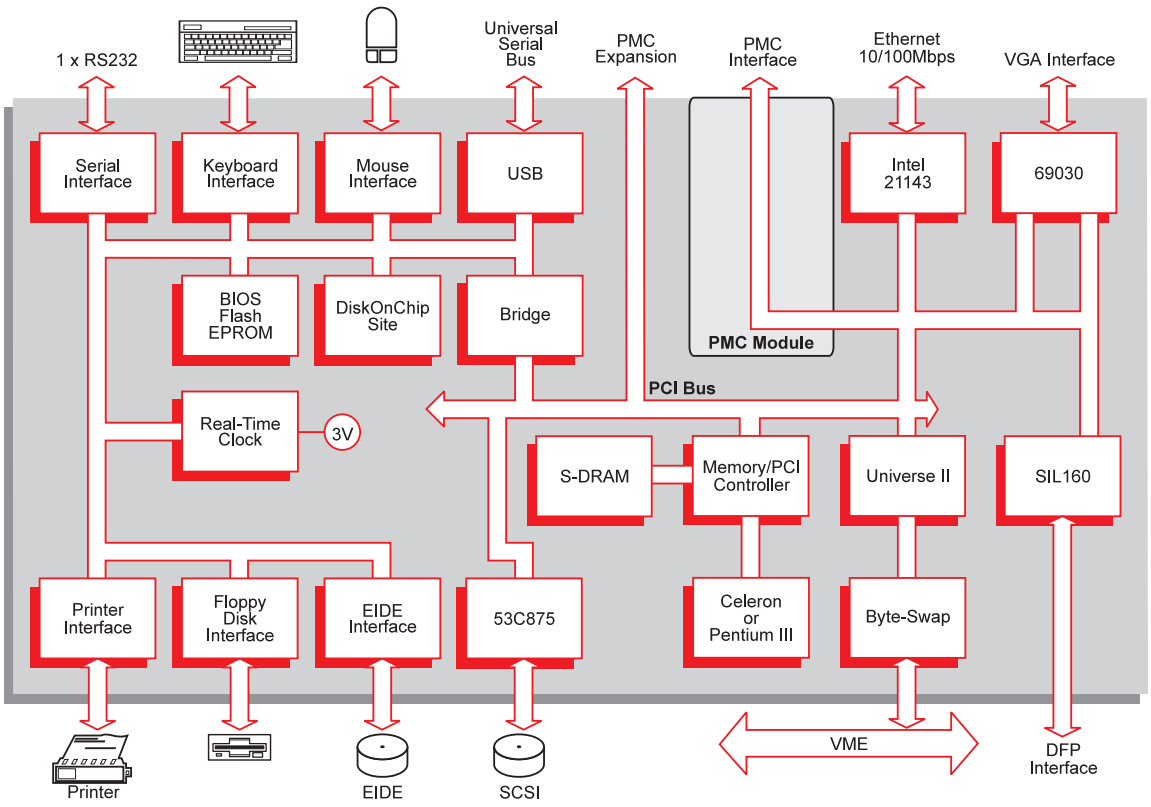


Figure 5-1 Overview

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

5.3 Memory Structure

Many areas of the memory and I/O maps are software configurable via registers which reside within the PCI devices configuration space. The PCI devices are the: 443BX memory controller, 53C875 SCSI controller, 21143 Ethernet controller, 91C142 VME interface, PIIX4E, 69030 Graphics Controller, PMC module and the PCI expansion card if present. PCI bus device number assignment is given in Appendix B. The format of the PCI configuration registers are shown in Figure 5-2, refer to the PCI specification revision 2.1 for further details. Within each device are a number of device specific registers which can affect the memory and I/O maps, refer to the device data sheets for further information. Details of maps provided in this chapter assume that the standard factory firmware is fitted to the board and the contents of PCI configuration registers have not been altered by user code.

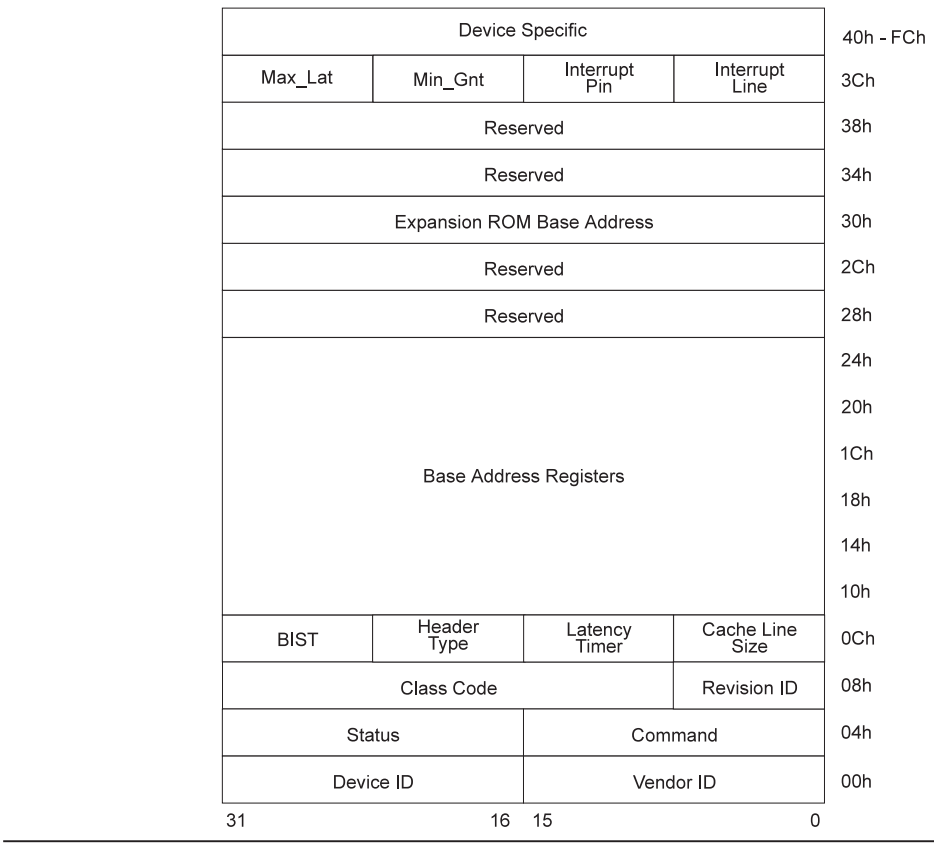


Figure 5-2 PCI Configuration Registers

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

5.3.1 Memory Maps

Above the first Mbyte is on board SDRAM which can be up to a further 383 Mbytes. Above the fitted SDRAM to the bottom of EPROM (FFF80000h) is PCI free memory or VME bus memory. Any PCI bus device requiring to be memory mapped can be mapped into this area. The PCI configuration register can be read to determine where a device has been mapped. Figure 5-3 shows the default address map for the board. Refer to Chapter 4 for examples of code to access the PCI registers.

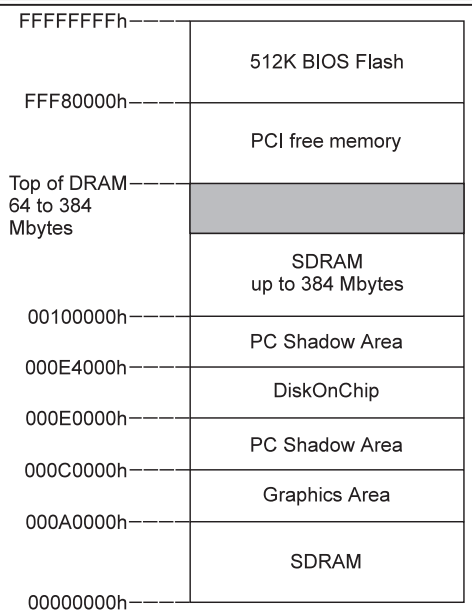


Figure 5-3 Memory Map

5.3.2 Memory Controller

The memory controller provides the SDRAM controller and PCI bus interface support.

The SDRAM controller supports 64 Mbytes or 128 Mbytes of soldered on SDRAM and 64 Mbytes, 128 Mbytes, or 256 Mbytes on a 144-pin SODIMM. The controller supports 66MHz (Celeron variants only) and 100MHz SDRAM devices, operating at 66MHz (Celeron variants) and 100MHz (Pentium III variants) respectively to achieve an in-page 32 byte burst read in 10 CPU bus clocks. To achieve higher performance the controller has a posted write buffer which reduces a 32 byte burst write to 6 clocks.

The memory controller provides CPU to PCI bus access support, and also supports accesses to SDRAM for PCI bus masters. To achieve maximum PCI bus performance the controller performs write posting and read ahead for PCI bus master accesses.

5.3.3 SDRAM

The board has 64 or 128 Mbytes of SDRAM soldered onto the board. In addition, provided on the board is a single 144 pin SODIMM socket which accept 3.3 Volt 144-pin 64 bit wide SDRAM memory modules. SODIMM sizes supported are 8M x 64, 16M x 64, and 32M x 64. The maximum memory that can be fitted is 384 Mbytes, consisting of 128 Mbytes soldered onto the board and a 256 Mbyte SODIMM module.

5.3.4 PC BIOS Flash EPROM

The PC BIOS Flash EPROM is a single JEDEC 32-pin PLCC socket which will accept a 5V only byte-wide device of 512 Kbytes. EPROM accesses do not support processor burst mode addressing, however if shadowed to SDRAM by the Memory Controller burst accesses to the same address range are supported.

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

5.3.5 Off-board VME Memory

The VP PSE/P3x can be programmed as a VME master supporting off-board VME memory addressing accessible by any PCI master.

The VP PSE/P3x can also be programmed as a VME slave allowing other VME masters to access any PCI slave.

This access is achieved by programming the appropriate Universe II device register, "PCI slave" registers for VP PSE/P3x VME master accesses and "VME slave" registers for VME accesses to the VP PSE/P3x.

The VME interface supports A32/A24/A16/MBLT64 addressing modes and D64/D32/D16/D08 (EO) data widths in both user and supervisor address space.

The VME interface performs auto-syscon detect at power-up to provide system controller functionality, if the board is located in the first VME slot. As system controller the Universe II will arbitrate VME mastership of the bus using DEMAND request mode.

The VP PSE/P3x can act as an interrupt controller for any combination of VME interrupts and can be an interrupter generating either a software interrupt or any of the Universe's internal interrupt sources on any IRQ level. All VME interrupts are directly mapped between the Universe II registers and the VME bus backplane. Of the PCI LINT lines only LINT0 is mapped into the PCI interrupt map and with LINT1 mapped to NMI (see sections 5.4.1.1 and 5.4.1.2).

The Universe II device uses the linear incrementing mode when being accessed by a PCI master.

The Universe II supports VME mailbox interrupts. See Universe II data sheet for further details.

WARNING VME bus access is allowed to the full VP PSE/P3x memory map. Care must be taken to ensure that no accesses are made to areas that will corrupt the system memory or the configuration of any of the interfaces.

5.3.6 VME Byte Swapping

The VP PSE/P3x provides hardware that performs fast byte swapping for aligned D16 and D32 VME transfers. Byte swapping can be enabled separately for master and slave transfers under software control. Swapping is performed as follows.

D16 (Double Byte 2 - 3):

```
D[31.....24]     < - >   D[23.....16]
D[23.....16]     < - >   D[31.....24]
```

D16 (Double Byte 0 - 1):

```
D[15....8]       < - >   D[7.....0]
D[7.....0]       < - >   D[15....8]
```

D32 (Quad Byte 0 - 3):

```
D[31....24]      < - >   D[7.....0]
D[23....16]      < - >   D[15....8]
D[15.....8]      < - >   D[23...16]
D[7......0]      < - >   D[31...23]
```

The hardware decodes the VME transfer taking place to see if it is swappable, checks to see if swapping is enabled and then configures a set of multiplexors to perform the required data swap. For master and slave read cycles the byte swap hardware imposes negligible delay on the VME bus cycle since the decode and configuration occur before the data is valid. For write cycles the hardware imposes an approximate delay of 50ns in order to provide the required data setup time before the data strobes are asserted. The delay applies to single cycle transfers and the first cycle of block transfers.

NOTE The delay can be turned off under software control, but only if the user can guarantee that only swappable cycles will be run across the VME interface.

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

5.4 I/O Functions

The VP PSE/P3x supports a number of peripherals and interfaces within the 64 Kbytes I/O address range, as shown in Figure 5-4, and in more detail in Table B-1. As with the memory map, software configurable PCI registers determine the I/O address of a number of the I/O functions. Details within this chapter and the addresses given in Appendix C are based on the factory standard firmware initialization. Examples of routines to access PCI registers are given in Chapter 4.

Recovery times for all the on-board I/O devices are satisfied automatically by the hardware.

5.4.1 PCI ISA IDE Xcelerater (PIIX4E)

Intel's PIIX4E device is used to provide the following functions:-

- ISA Bus controller with integrated DMA, timer and interrupt controller;
- Real time clock with 256 bytes of battery backed CMOS RAM;
- USB Controller; and
- Enhanced IDE controller with PCI and Ultra DMA-33 extensions.

Channel 2 of the DMA controller is used for floppy disk data transfers to SDRAM. Refer to device data book for further details. The following sections provide VP PSE/P3x specific information.

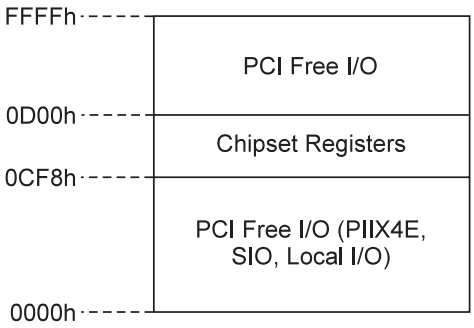


Figure 5-4 I/O Space Map

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

5.4.1.1 Interrupt Controllers

The PIIX4E contains a 8259 compatible master and slave PIC to provide 15 interrupt levels, assignments are shown in Table 5-1.

Controller	Request No.	Priority	Interrupt Source
Master	IRQ0	1	Timer 0
Master	IRQ1	2	Keyboard
Master	IRQ2	3-10	Slave PIC
Slave	IRQ8	3	Real Time Clock / Alarm
Slave	IRQ9	4	Reserved
Slave	IRQ10	5	Reserved
Slave	IRQ11	6	Reserved
Slave	IRQ12	7	Mouse
Slave	IRQ13	8	FPU Error
Slave	IRQ14	9	EIDE
Slave	IRQ15	10	Reserved
Master	IRQ3	11	Serial Channel 1(Not Used)
Master	IRQ4	12	Serial Channel 0
Master	IRQ5	13	LPT2
Master	IRQ6	14	Floppy Disk
Master	IRQ7	15	LPT1

Table 5-1 Interrupt Assignments

Master PIC IRQ0 - TIMER 0
The 8254 compatible timer device has timer 0 output assigned to IRQ0.

Master PIC IRQ1 - KEYBOARD
This interrupt is connected to the keyboard controller.

Master PIC IRQ2 - SLAVE PIC
The slave PIC is cascaded on master PIC level 2.

Master PIC IRQ3 - Serial Channel 1
This interrupt is assigned to serial channel 1. The Super I/O Controller must also be correctly configured for this assignment. This serial channel is not used.

Master PIC IRQ4 - Serial Channel 0
This interrupt is assigned to serial channel 0. The Super I/O Controller must also be correctly configured for this assignment.

Master PIC IRQ5 - LPT2
This interrupt may be assigned to the parallel port. The Super I/O Controller must also be correctly configured for this assignment. Typically IRQ 5 is not used.

Master PIC IRQ6 - Floppy Disk
This interrupt is routed to the Super I/O Controller floppy disk interrupt.

Master PIC IRQ7 - LPT1
Alternative parallel port interrupt. The Super I/O Controller must also be correctly configured for this assignment.

Slave PIC IRQ8 - RTC
This interrupt is assigned to the Real Time Clock device.

Slave PIC IRQ9 - Reserved
Reserved interrupts may be used as PCI interrupts routed via the PIRQ Routing Control register within the PIIX4E.

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

Slave PCI IRQ10 - Reserved
Reserved interrupts may be used as PCI interrupts routed via the PIRQ Routing Control register within the PIIX4E.

Slave PIC IRQ11 - Reserved
Reserved interrupts may be used as PCI interrupts routed via the PIRQ Routing Control register within the PIIX4E.

Slave PCI IRQ12 - Mouse
This interrupt is connected to the mouse controller.

Slave PIC IRQ13 - FERR
This is connected to the floating point unit error signal. The FPU is within the processor device.

Slave PIC IRQ14 - EIDE
This interrupt is connected to the EIDE interface.

Slave PCI IRQ15 - Reserved
Reserved interrupts may be used as PCI interrupts routed via the PIRQ Routing Control register within the PIIX4E.

5.4.1.2 PCI Interrupts

There are four PCI interrupts which are active low level sensitive signals.

The interrupts are connected to the following devices:

PIRQ0	Ethernet, PCI Expansion, PMC INTD
PIRQ1	Graphics, PCI Expansion, PMC INTA
PIRQ2	SCSI 53C875, PCI Expansion, PMC INTB
PIRQ3	Universe LINT0, PCI Expansion, PMC INTC

These are wired OR interrupt lines.

5.4.1.3 NMI

The processor's NMI signal can be generated from four sources: Watchdog timeout, Universe LINT1, SERR or front panel switch.

SERR is controlled via the PIIX4E.

Watchdog timeout, LINT1 and the front panel switch generate NMI via IOCHK to the PIIX4E. To use these services IOCHK must be enabled in the NMI Mask register in the PIIX4E. LINT1 is sourced and controlled via the Universe II. The status of these lines are reported in On-Board Status/Control Register 0 and the Watchdog Status & Control Register. The Watchdog must be jumper enabled and software configured. The front panel switch must be jumper enabled.

5.4.1.4 Timers

An 8254 compatible Programmable Interval Timer (PIT) is provided in the PIIX4E. Timer 0 has no board specific functions, timer 2 sources the speaker output which drives the POST LED. Their outputs are connected to master PIC IRQ0 and Slave PIC IRQ12 respectively. Timer 1 is not used. The clock source for the timers is a continuous frequency of 1.193MHz. The gate input for timers 0 and 1 are permanently held in an active state. Gate input for timer 2 is programmable via I/O port 61h, NMI status and control register within the PIIX4E.

5.4.1.5 EIDE Interface

The primary EIDE interface on the PIIX4E is routed to the P2 connector. Two variants of the P2 pin assignment are supported. One version has the EIDE interface on the outer two rows of the 160-way P2 connector, with SCSI provided on the inner rows. The other version has the EIDE interface on both the outer and inner rows, SCSI is not supported. This second version is intended for EIDE only applications on a 96-way P2 backplane. Refer to Appendix A for pin out details. Breakout boards are available to provide a 40-way IDC connector.

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

5.4.2 Super I/O Controller

The National Semiconductor Super I/O Controller device provides the standard PC-AT peripheral interfaces.

The super I/O device contains two 16550 compatible UARTs, one of which is used to provide a standard asynchronous serial port with RS232 interface levels.

The parallel port can be configured as one of four PC standards, ISA-compatible, PS/2-compatible, EPP and ECP. It supports five interface protocols as described in IEEE1284.

The serial port is available via a 9-way D-type connector on the front panel. The floppy disk interface is available via the P2 connector. A breakout board is available to provide a 34-way IDC connector.

A 24MHz clock is divided down by the Super I/O Controller to provide a 1.8462MHz clock for the internal baud rate generation.

5.4.3 Real Time Clock (RTC)

The PIIX4E provides standard PC-AT clock and calendar functions. The device has an onboard battery to provide data retention in the absence of power to the board. Included in the device are 256 bytes of general purpose RAM which is also battery backed. The RAM is used to store standard PC-AT setup information plus other board parameter information.

5.4.4 Keyboard & Mouse Controllers

An 82C42PC controller provides standard Keyboard and Mouse interfaces. The controller supports all standard PC I/O port 60h and 64h functions. Refer to the 82C42PC data manual for further details.

The Keyboard interface is accessible via a front panel mounted 6 pin mini-DIN connector which can support a PS/2 keyboard or PC-AT compatible keyboard with a PS/2 adapter cable.

The Mouse interface is accessible via a front panel mounted 6 pin mini-DIN connector which can support a PS/2 mouse.

Power for the keyboard and mouse interfaces are protected by self resetting 0.75A current limiting circuits.

5.4.5 Ethernet Controller

The Intel 21143 Ethernet controller is used to provide a high performance PCI to Ethernet interface. A category 5 RJ45 interface is provided via the front panel. Refer to the device data sheet and chapter 4 of this manual for programming information.

The interface can support both 10 Base-T and 100Base-TX and is compliant with IEEE 802.3 standards.

5.4.6 Graphics Interface

The graphics interface utilizes the highly integrated CHIPS 69030 device to provide a VGA interface.

The 69030 incorporates 4 Mbytes of SDRAM for the graphics/video frame buffer, and also contains a flat panel display controller.

5.4.7 Flat Panel Interface

The board can be configured to provide a DFP compatible flat panel interface via P2. The interface uses the flat panel controller in the graphics device and a SIL160A TMDS transmitter. A breakout board is available to provide a 20-way DFP connector.

5.4.8 DiskOnChip

A 32 pin DIL socket is provided for the user to install a DiskOnChip device. Any capacity of device may be installed. Pin 1 of the socket is nearest to the front panel mouse connector and is shown in Figure A-1.

NOTE The BIOS places restrictions on the use of DiskOnChip in conjunction with the SCSI controller. See Section 6.2.5 for details.

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

5.4.9 SCSI Controller

The VP PSE/P3x board uses a 53C875 Ultra SCSI I/O Controller to provide a 16-bit SCSI interface, via the P2 connector except on EIDE only versions of the boards. Breakout boards are available to provide a 50-way IDC connector, for Narrow SCSI or a 68-way high density connector for Wide SCSI. The 53C875 is a very powerful device containing its own DMA controller and SCRIPT processor enabling high performance and ease of programming. For further details see the relevant data sheet and programming examples in Chapter 4.

The SCSI interfaces are terminated on board with active termination devices to support the maximum Ultra SCSI 16 bit bus data transfer rate of 40 Mbytes/s on a single ended interface. The termination devices can be permanently enabled or disabled.

The SCSI terminators are protected by a self resetting 1A current limiting circuit.

NOTE The BIOS places restrictions on the use of DiskOnChip in conjunction with the SCSI controller. See Section 6.2.5 for details.

5.4.10 Processor Debug Port

A Processor Debug Port which supports the In Target Probe (ITP) for software development is provided. The port interface connector is a 30-way socket which complies with the 1.5V ITP requirements defined by Intel, the location and pin 1 indication is shown on Figure A-1.

This socket is compatible with ITP's for Celeron 533MHz and Pentium III 700 or 850MHz processors. When using the port for a Celeron 433MHz processor an adaptor is required, consult Concurrent Technologies for further details.

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

5.5 443BX Chipset Registers

Intel's 443BX Memory Controller, (North Bridge), has many internal control registers which are in PCI configuration address space. A brief description of the permanent registers I/O Address Space is given below. Refer to the data sheet for further details.

5.5.1 Configuration Space Enable Register/Configuration Address Register

This register enables or disables PCI configuration space accesses. It also contains the bus number, device number, functional number and register number for the intended configuration access.

Only PCI Configuration Access Mechanism 1 is supported on this product. With access mechanism 1, PCI configuration space is accessed using Configuration Addres and Data Registers. When writing to the Configuration Address Register 32-bit accesses must be used.

5.5.2 Configuration Data Register

This register is used to perform PCI configuration space accesses using access mechanism 1.

5.5.3 PCI Arbiter

The 443BX provides a PCI arbiter which supports the seven potential PCI bus masters with the assignments shown in Table 5-2. The arbiter implements a round robin priority scheme between all potential masters.

443BX Pin Name	Device Connected
N/A	Internal request (CPU)
PHOLD	PIIX4E
REQ 0	Ethernet Controller
REQ 1	SCSI Controller
REQ 2	VME Bridge
REQ 3	PCI Expansion interface
REQ 4	PMC interface

Table 5-2 PCI Arbiter Priority Assignment

NOTE PCI Bus parking function is performed by the 443BX.

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

5.6 Front Panel LEDs

The VP PSE/P3x is equipped with a pushbutton switch and four LEDs on the front panel to control and monitor the current status of the board. The four LEDS have the following functions:

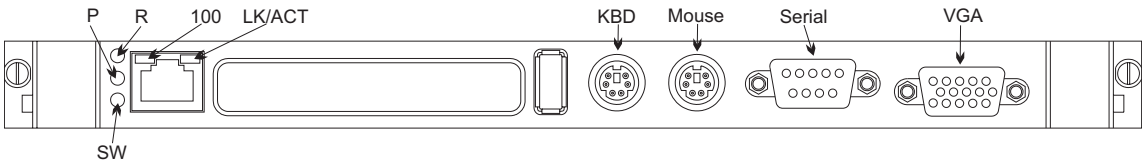


Figure 5-5 Front Panel LEDs

5.6.1 Run LED (R) Green

The run LED indicates that activity is occurring on the PCI bus. This allows the user to quickly assess how active the PCI bus is.

5.6.2 POST LED (P) Yellow

The POST LED is used to indicate that a power on self test has failed. The LED is controlled by the speaker port. The LED will flash during the memory test after a reset. See section 6.3 for details.

5.6.3 100 Mbit Enabled (100) Yellow

The 100 LED lights when the Ethernet interface is running at 100 Mbit/s. The LED will go out when running at 10 Mbit/s.

5.6.4 Link/Activity LED (LK/ACT) Green

Lights when connection has been made. Flashes to indicate link activity, during periods of high Ethernet activity this LED may go out for several seconds.

5.6.5 Switch (SW)

The switch button can be used for generating system reset, local reset or NMI.

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

This page has been left intentionally blank

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

PC BIOS Firmware

6.1 General

The VP PSE/P3x board is supplied with PhoenixPICO BIOS™ Release 4.0. Also included are Symbios Logic SCSI BIOS extensions and Intel (Chips and Technologies) 690x0 Video BIOS extensions.

6.2 Setup Options

With the PhoenixPICO BIOS Setup program, you can modify BIOS settings and control the special features of the system. The Setup program uses a number of menus for making changes and turning the special features on or off.

The menus shown here are from a typical VP PSE/P3x board. The actual menus displayed on your screen depend on the specific hardware and features available.

The PhoenixPICO BIOS Setup utility is entered by pressing <F2> whenever the BIOS displays this message:

```
Press <F2> to enter SETUP
```

6.2.1 Menu Bar

The Menu Bar at the top of the window lists these selections:

Use the left/right “ ” arrow keys to make a selection.

Main	Use this menu for basic system configuration.
Advanced	Use this menu to set the Advanced Features available on your system’s chipset.
Security	Use this menu to set User and Supervisor Passwords and the Backup and Virus-Check reminders.
Universe	Use this menu to configure the Universe PCI – VME bridge.
Boot	Use this menu to configure the boot strategy.
Exit	Exits the current menu.

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

6.2.2 Legend Bar

Use the keys listed in the legend bar on the bottom to make your selections or exit the current menu. The chart below describes the legend keys and their alternates:

Key	Meaning
<F1> or <Alt-H>	General Help window.
<Esc>	Exit this menu.
← or → arrow keys	Select a different menu.
↑ or ↓ arrow keys	Move cursor up and down.
<Tab> or <Shift-Tab>	Cycle cursor up and down.
<Home> or <End>	Move cursor to top or bottom of window.
<PgUp> or <PgDn>	Move cursor to next or previous page.
<F5> or <->	Select the Previous Value for the field.
<F6> or <+> or <Space>	Select the Next Value for the field.
<F3>	Load the Default Configuration values for this menu.
<F4>	Load the Previous Configuration values for this menu.
<Enter>	Execute Command or Select a Sub-menu.
<Alt-R>	Refresh screen.

To select an item, use the arrow keys to move the cursor to the field you want. Then use the plus-and-minus value keys to select a value for that field. The Save Values commands in the Exit Menu save the values currently displayed in all the menus.

To display a sub menu, use the arrow keys to move the cursor to the sub menu you want. Then press <Enter>.

A "»" pointer marks all sub menus.

6.2.3 Field Help Window

The help window on the right side of each menu displays the help text for the currently selected field. As you move the cursor to each field, it updates the values.

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

6.2.4 General Help Window

Pressing <**F1**> or **<Alt-H>** on any menu brings up the General Help window that describes the legend keys and their alternates:

```
General Help
 Setup changes system behavior by modifying the power on
 initialization parameters. Selecting incorrect values
 may cause system boot failure; load Setup Default values
 to recover.

 <Up/Down> arrows select fields in current menu.
 <PgUp/PgDn> moves to previous/next page on scrollable
 menus.
 <Home/End> moves to top/bottom item of current menu.

 Within a field,<F5> or <-> selects next lower value and
 <F6>, <+>, or <Space> selects next higher value.
 <Left/Right> arrows select menus on menu bar.
 <Enter> displays more options for items marked with >.

 <F3>loads factory-installed Setup Default values.
 <F4>restores previous values from CMOS.

 <ESC> or <Alt-X> exits Setup; in sub-menus, pressing
 these keys returns to the previous menu.

 <F1> or <Alt-H> displays General Help (this screen).

                            [Continue]
```

The scroll bar on the right of any window indicates that there is more than one page of information in the window. Use **<PgUp>** and **<PgDn>** to display all the pages. Pressing **<Home>** and **<End>** displays the first and last page. Pressing **<Enter>** displays each page and then exits the window.

Press **<Esc>** to exit the current window.

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

6.2.5 Main Setup Menu

The Main Setup Menu provides basic system configuration options: system time and date, floppy, fixed and ROM disk settings and memory configuration.

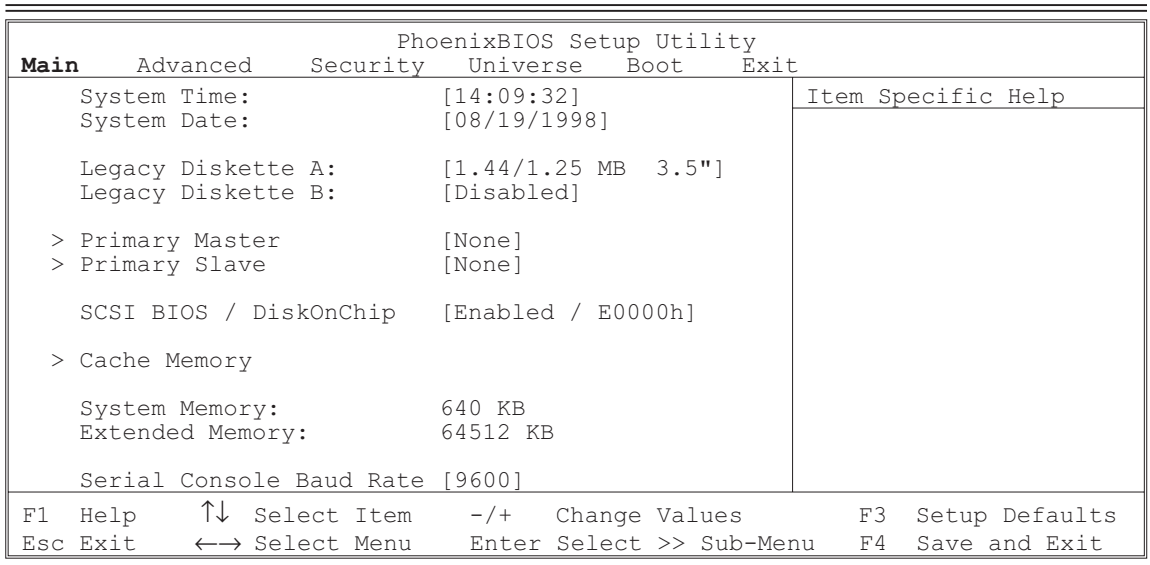


Figure 6-1 The Main Setup Menu

Feature	Options	Description
System Time	HH:MM:SS	Set the system time.
System Date	MM/DD/YYYY	Set the system date.
Disk A: Disk B:	Not installed 360KB, 5 ¼“ 1.2MB, 5 ¼“ 720KB, 3 ½“ 1.44MB, 3 ½” 2.88MB, 3 ½”	Select the type of floppy-disk drive installed in your system.
Primary Master	(See sub-menu)	Configures the master IDE drive on the primary controller.
Primary Slave	(See sub-menu)	Configures the slave IDE drive on the primary controller.
SCSI BIOS / DiskOnChip	Enabled / E0000h Disabled / E0000h Disabled / CC000h	Enables / Disables the SCSI BIOS and selects the base address for the DiskOnChip BIOS. (See NOTES on the following page).
Cache Memory	(See sub-menu)	Determines how to configure the specified block of memory.
System Memory	N/A	Displays amount of conventional memory detected during boot-up.
Extended Memory	N/A	Displays the amount of extended memory detected during boot-up.
Serial Console Baud Rate	600 1200 2400 4800 9600 19.2K 38.4K 115.2K	Enables the specified baud rate for the serial console.

Table 6-1 Main Setup Options

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

NOTE The SCSI BIOS can not be enabled when DiskOnChip is at CC000h.

NOTE Windows NT will not work with DiskOnChip at E0000h.

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

6.2.5.1 Main: Primary Master and Primary Slave Sub-Menus

The VP PSE/P3x includes one IDE adapter; this supports one master drive and one optional slave drive.

These sub-menu setup screens allow the user to select the type of disk connected to each of the IDE controller drives. The BIOS can automatically detect disk drive parameters, or the user may specify them manually.

In the automatic modes, the Cylinder, Head and Sector parameters are not displayed.

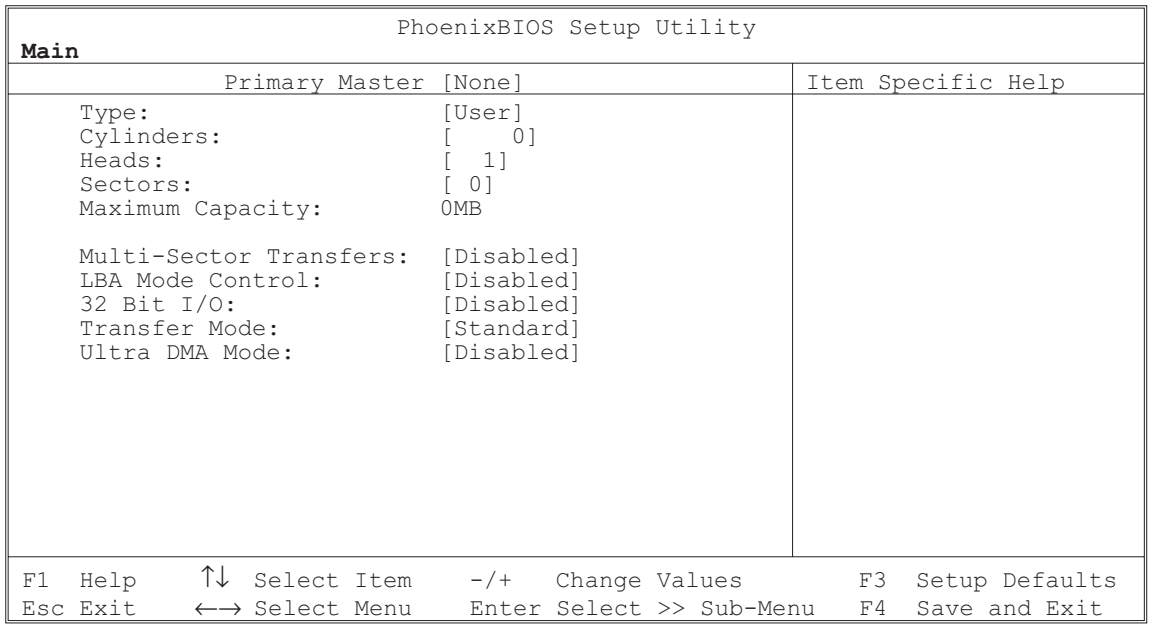


Figure 6-2 Primary Master / Primary Slave Sub-Menu

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

Feature	Options	Description
Type	None CD-ROM ATAPI Removable User Auto	No drive / Disable the drive completely IDE CD-ROM IDE Hard disk, user sets parameters IDE Hard disk, BIOS sets parameters
Cylinders	1 – 16,384	Cylinder count
Heads	1 – 16	Head count
Sectors	1 – 63	Sectors per track
Maximum Capacity	N/A	Displays the maximum capacity for a drive with the specified parameters
Multi-Sector Transfers	Disabled 2 Sectors 4 Sectors 8 Sectors 16 Sectors	Auto sets the number of sectors per block at the highest number supported by the drive. This is not always the fastest option.
LBA Mode Control	Disabled Enabled	Enables Logical Block Addressing mode
Ultra DMA Mode	Disabled Mode 0 Mode 1 Mode 2	Selects the DMA transfer mode when supported
32 Bit I/O	Disabled Enabled	Enables 32-bit I/O transfers
Transfer Mode	Standard Fast PIO 1 Fast PIO 2 Fast PIO 3 Fast PIO 4 FPIO 3 / DMA 1 FPIO 3 / DMA 2	Selects the method of transfering the data between the hard disk and system memory. The Setup menu only lists those options supported by the drive and platform

Table 6-2 Primary Master / Primary Slave Options

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

6.2.5.2 Main: Memory Cache Sub-Menu

This sub-menu provides control over the Level 2 cache.

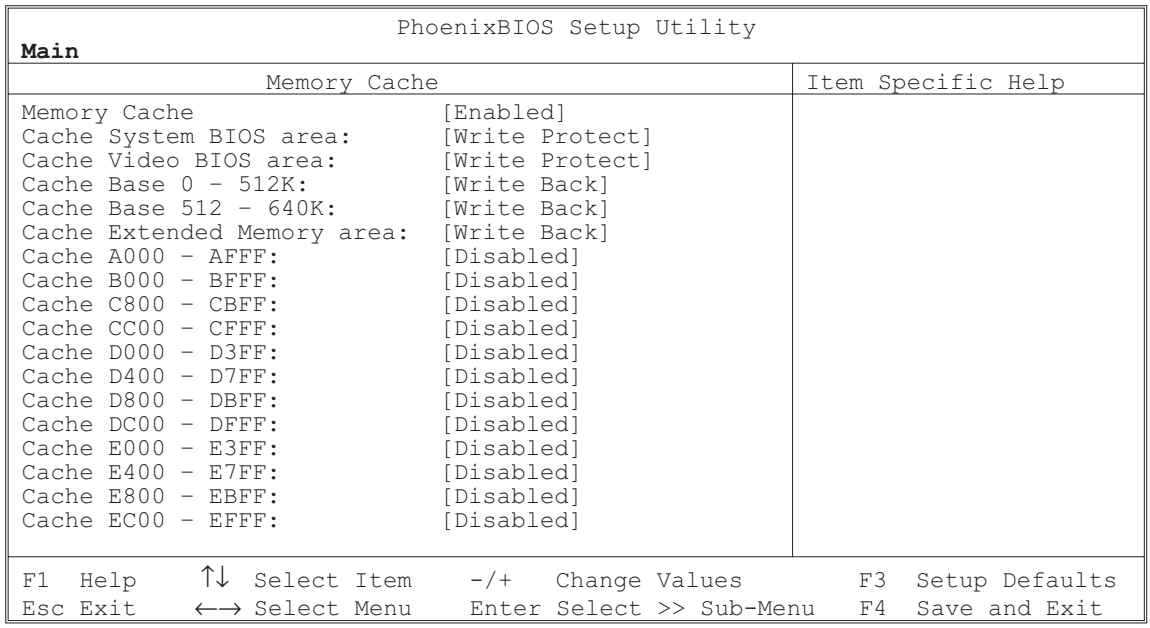


Figure 6-3 Memory Cache Sub-Menu

Feature	Options	Description
Memory Cache	Disabled Enabled	Disable Level 2 cache
Cache System BIOS area	Uncached Write Protect	No caching of system BIOS System BIOS cached and write protected
Cache Video BIOS area	Uncached Write Protect	No caching of video BIOS Video BIOS cached and write protected
Cache Base 0 – 512K	Uncached Write Through Write Protect Write Back	Base 0 – 512K memory uncached Caching policy
Cache Base 512 – 640K	Uncached Write Through Write Protect Write Back	Base 512 – 640K memory uncached Caching policy
Cache Extended Memory area	Uncached Write Through Write Protect Write Back	Extended memory uncached Caching policy
Cache A000 - BFFF	Disabled USWC caching Write Through Write Protect Write Back	Cache disabled Caching policy

Table 6-3 Memory Cache Options

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

Feature	Options	Description
Cache B000 - BFFF	Disabled USWC caching Write Through Write Protect Write Back	Extended memory uncached Caching policy
Cache CC00 – CFFF	Disabled USWC caching Write Through Write Protect Write Back	Extended memory uncached Caching policy
Cache D000 – D3FF	Disabled USWC caching Write Through Write Protect Write Back	Extended memory uncached Caching policy
Cache D400 – D7FF	Disabled USWC caching Write Through Write Protect Write Back	Extended memory uncached Caching policy
Cache D800 – DBFF	Disabled USWC caching Write Through Write Protect Write Back	Extended memory uncached Caching policy
Cache DC00 – DFFF	Disabled USWC caching Write Through Write Protect Write Back	Extended memory uncached Caching policy
Cache E000 – E3FF	Disabled USWC caching Write Through Write Protect Write Back	Extended memory uncached Caching policy
Cache E400 – E7FF	Disabled USWC caching Write Through Write Protect Write Back	Extended memory uncached Caching policy
Cache E800 – EBFF	Disabled USWC caching Write Through Write Protect Write Back	Extended memory uncached Caching policy
Cache EC00 – EFFF	Disabled USWC caching Write Through Write Protect Write Back	Extended memory uncached Caching policy

Table 6-3 Memory Cache Options (continued)

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

6.2.6 Advanced Setup Menu

This menu provides the user with access to specific features of the board's chipset. These can be modified to optimize the performance of the board.

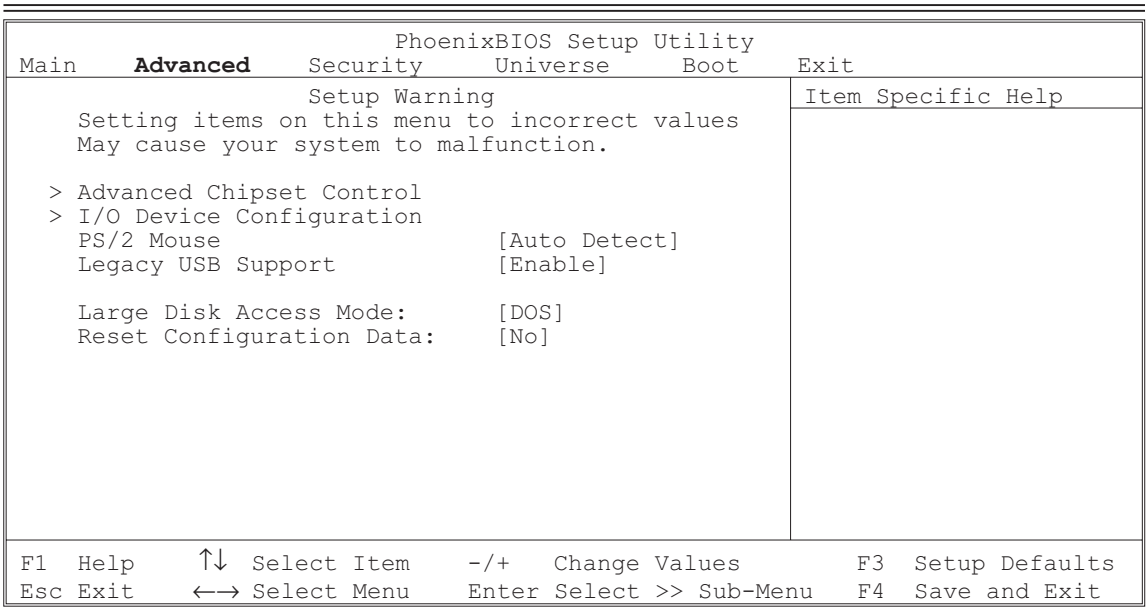


Figure 6-4 Advanced Setup Menu

Feature	Options	Description
Advanced Chipset Control	(see sub-menu)	Sub-menu for tuning chipset performance
I/O Device Configuration	(see sub-menu)	Sub-menu providing configuration of I/O peripherals
PS/2 Mouse	Disabled Enabled Auto Detect	Disables any installed mouse, also frees IRQ12 Forces the interface enabled even with no mouse is installed Enables the interface only if a mouse is installed
Legacy USB Support	Enable Disable	Enable USB controller Disable USB controller
Large Disk Access Mode	DOS Other	When using UNIX, Novell or with other operating systems when problems are encountered during installation.
Reset Configuration Data	No Yes	Clear the Extended System Configuration Data (ESCD) area

Table 6-4 Advanced Setup Options

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

6.2.6.1 Advanced: Advanced Chipset Control

This sub-menu provides control over specific chipset features, which can be tuned to optimize the performance of the board. Refer to the appropriate chipset data manual for full details of the operation of these functions.

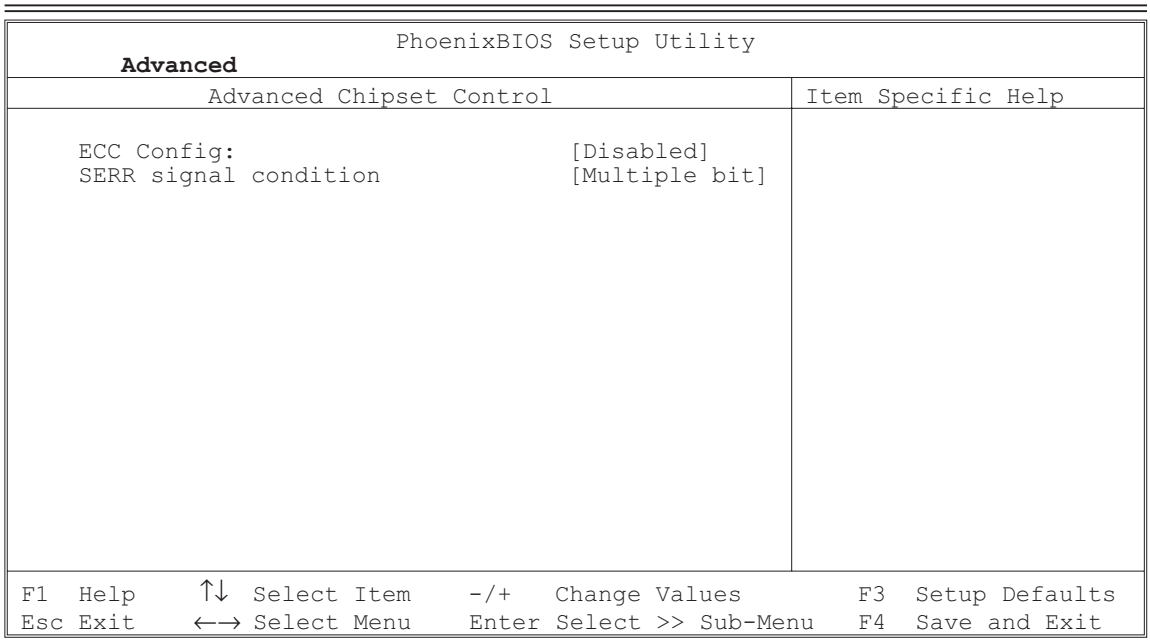


Figure 6-5 Advanced Chipset Control Sub-Menu

Feature	Options	Description
ECC Config	Disabled EC ECC ECC Scrub	Disable ECC checking
SERR signal condition	None Single bit Multiple bit Both	SERR not asserted on error SERR asserted for single bit errors SERR asserted for multiple bit errors SERR asserted for single and multiple bit errors

Table 6-5 Advanced Chipset Control Options

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

6.2.6.2 Advanced: I/O Device Configuration

This sub-menu provides control over the settings for the embedded parallel port and floppy disk controller.

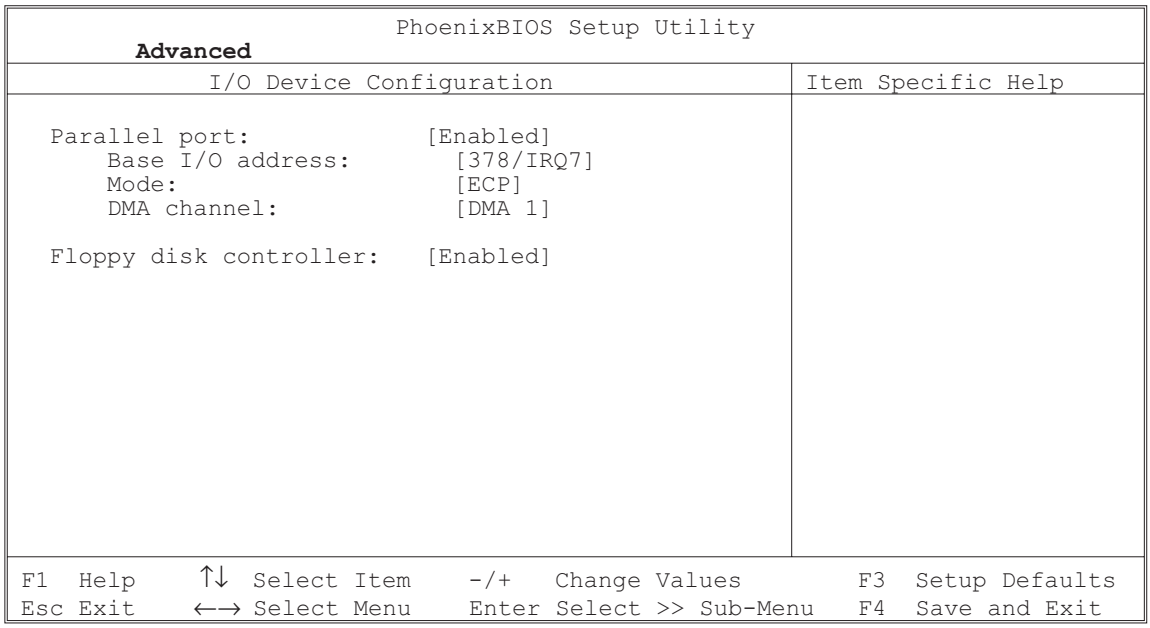


Figure 6-6 I/O Device Configuration Sub-Menu

Feature	Options	Description
Parallel Port	Disabled Enabled Auto	No configuration User specified Configuration BIOS or OS chooses configuration
Base I/O Address	3BCh / IRQ7 3BCh / IRQ5 378h / IRQ7 378h / IRQ5 278h / IRQ7 278h / IRQ5	I/O address and interrupt request combination
Mode	Output only Bi-directional ECP EPP	Printer port operational mode
DMA Channel	DMA 1 DMA 2 DMA 3	DMA channel used by printer interface
Floppy Disk Controller	Disabled Enabled Auto	No configuration User specified Configuration BIOS or OS chooses configuration

Table 6-6 I/O Device Configuration Options

NOTE Parallel port setup fields are only displayed when applicable to the selected operating mode.

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

6.2.7 Security Setup Menu

This menu provides password access control for both **user** and **supervisor** levels. Options are also included for displaying warning messages when Anti-Virus or System Backup software should be run.

The supervisor password must be set before a user password is configured. Once the supervisor password is configured, the user password gives restricted access to this Setup Utility.

If the supervisor password has been set, and password entry on boot is disabled, the BIOS operates with user level privileges.

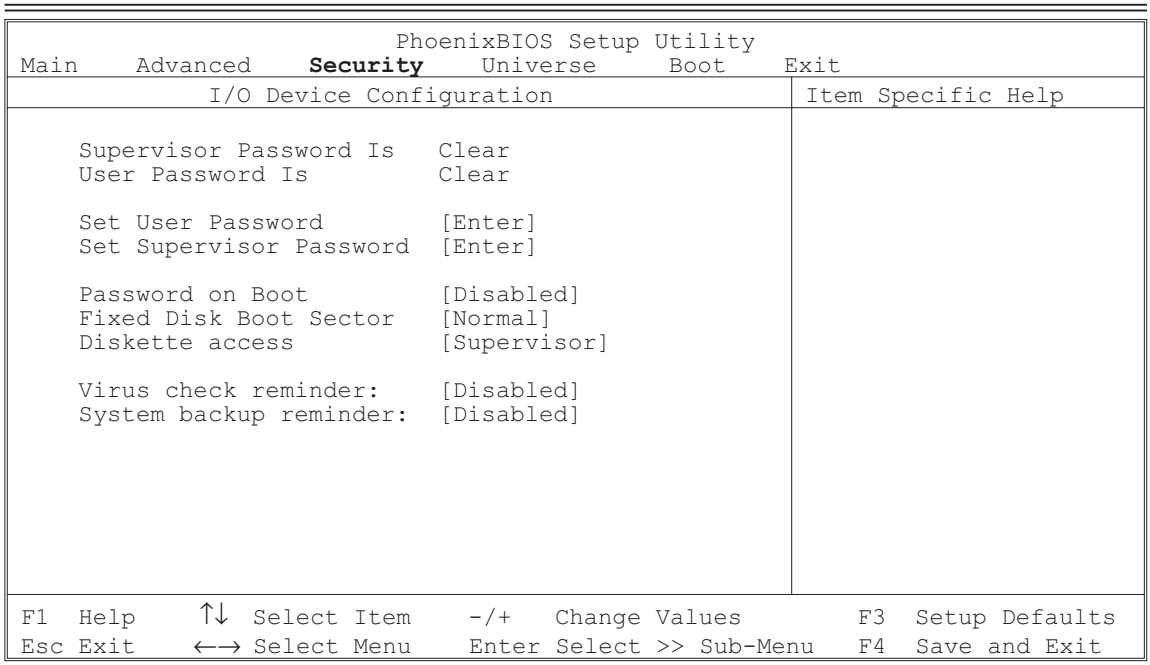


Figure 6-7 Security Setup Menu

Feature	Options	Description
Set User Password	N/A	Pressing <Enter> displays dialog box for entering the supervisor password. This password gives full access to SETUP menus.
Set Supervisor Password	N/A	Pressing <Enter> displays the dialog box for entering the user password. This password gives restricted access to SETUP menus. Requires prior setting of Supervisor password.
Virus Check Reminder	N/A	Enable a daily, weekly or monthly reminder to run virus check software.
System Backup Reminder	Disabled Enabled	Enable a daily, weekly or monthly reminder to run virus check software.
Password on boot	Disabled Enabled	Enabled requires a password on boot. Requires prior setting of the Supervisor password. If supervisor password is set and this option disabled, BIOS assumes user is booting.
Fixed disk boot sector	Normal Write Protect	Write protects the boot sector on the hard disk to protect against viruses
Diskette access	User Supervisor	Controls access to diskette drives.

Table 6-7 Security Setup Options

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

6.2.8 Universe Setup Menu

This menu controls the default configuration of the Universe II PCI-VME Bridge. The PCI slave and VME slave sub-menus each contain a scrolling setup list that provides configuration fields for the first four images of that type. Refer to the Universe II data book for full details of these options.

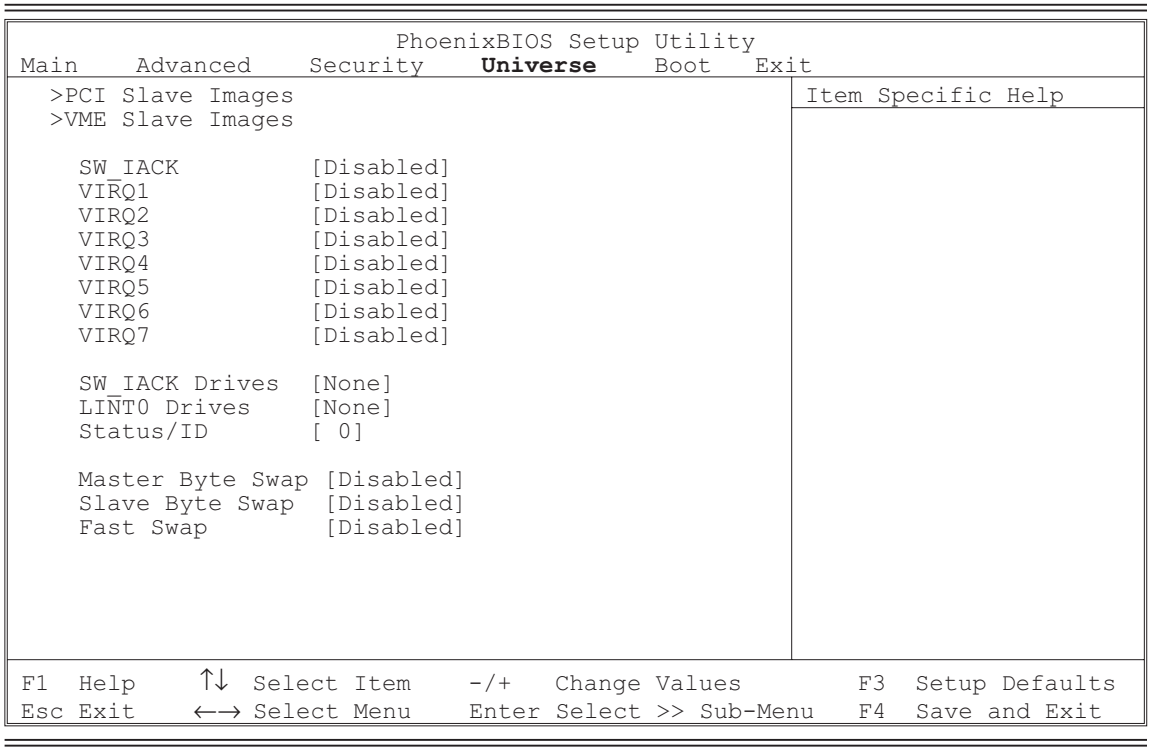


Figure 6-8 Universe Setup Menu

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

Feature	Options	Description
PCI Slave Images	(See sub-menu)	Configure the PCI Slave Images.
VME Slave Images	(See sub-menu)	Configure the VME Slave Images.
SW_IACK VIRQ1 VIRQ2 VIRQ3 VIRQ4 VIRQ5 VIRQ6 VIRQ7	Disabled Enabled	Specifies whether a VME bus interrupt source can drive the PCI-bus LINT0.
SW_IACK Drives	VINT1 VINT2 VINT3 VINT4 VINT5 VINT6 VINT7	Specifies which VME bus interrupt is driven by SW_IACK.
LINT0 Drives	(See above)	Specifies which VME bus interrupt is driven by LINT0
Status/ID	0 - 255	Specifies the 8-bit value returned by the Universe II during a VME-bus interrupt acknowledge cycle.
Master Byte Swap	Disabled Enabled	For aligned 16-bit transfers, D[15....8] is swapped with D[7....0]. For aligned 32-bit transfers, D[31....24] is swapped with D[7....0], and D[23....16] is swapped with D[15....8].
Slave Byte Swap	Disabled Enabled	For aligned 16-bit transfers, D[15....8] is swapped with D[7....0]. For aligned 32-bit transfers, D[31....24] is swapped with D[7....0], and D[23....16] is swapped with D[15....8].
Fast Swap	Disabled Enabled	Forces swap on ALL cycles by disabling the decode on VME cycles.

Table 6-8 Universe Configuration Options

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

6.2.8.1 Universe: PCI Slave Images

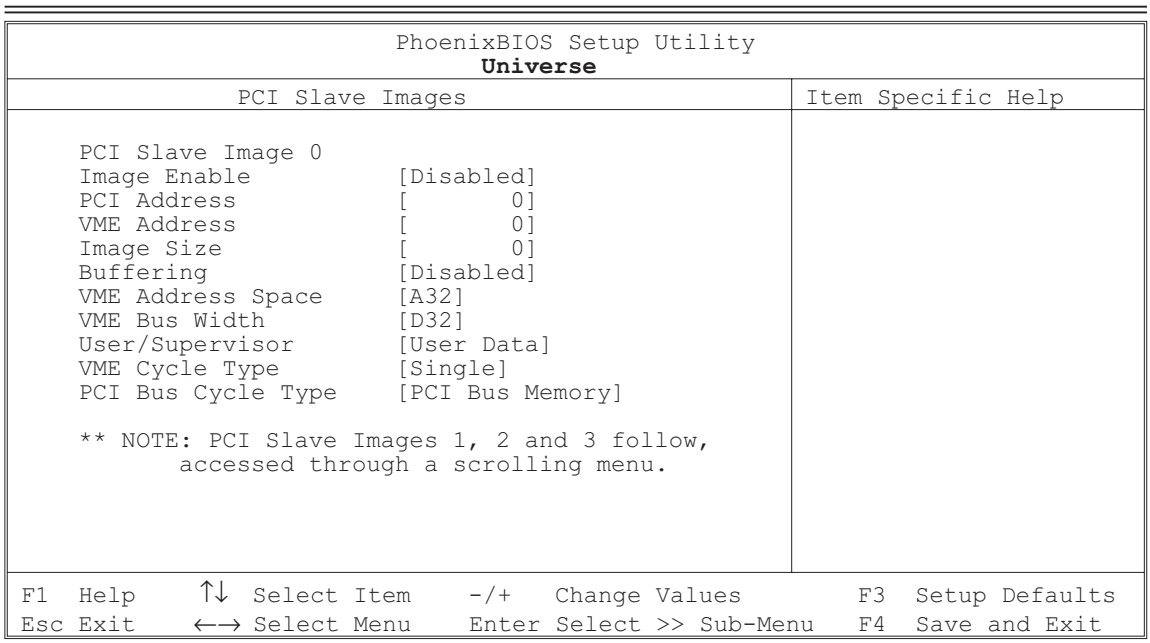


Figure 6-9 PCI Slave Images Sub-Menu

Feature	Options	Description
Image Enable	Disabled Enabled	Enable or Disable this slave image.
PCI Address	(32-bit address)	Hexadecimal PCI address for image. For image 0 this value is masked with 0xFFFFF000. For other images the value is masked with 0xFFFF0000.
VME Address	(32-bit address)	Hexadecimal VME address for image. For image 0 this value is masked with 0xFFFFF000. For other images the value is masked with 0xFFFF0000.
Image Size	(32-bit size)	Hexadecimal size of PCI/VME window to open in bytes. For image 0 this image is masked with 0xFFFFF000. For other images the value is masked with 0xFFFF0000.
Buffering	Disabled Enabled	Select Buffering mode. Enabling Buffering enables the Universe Posted Writes.
VME Address Space	A16 A24 A32 CR/CSR User 1 User 2	Select VME bus address space.
VME Bus Width	D8 D16 D32 D64	Select maximum VME bus width.
User Supervisor	User Data Supervisor Data User Program Supervisor Program	Select User or Supervisor, Program or Data access Address Modifier code.
VME Cycle Type	Single Block	Selects either single cycle or block transfers on the VME bus.
PCI Bus Cycle Type	PCI Bus Memory PCI Bus I/O PCI Bus config	Select PCI Bus Cycle type for transfer.

Table 6-9 PCI Slave Images Options

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

6.2.8.2 Universe: VME Slave Images

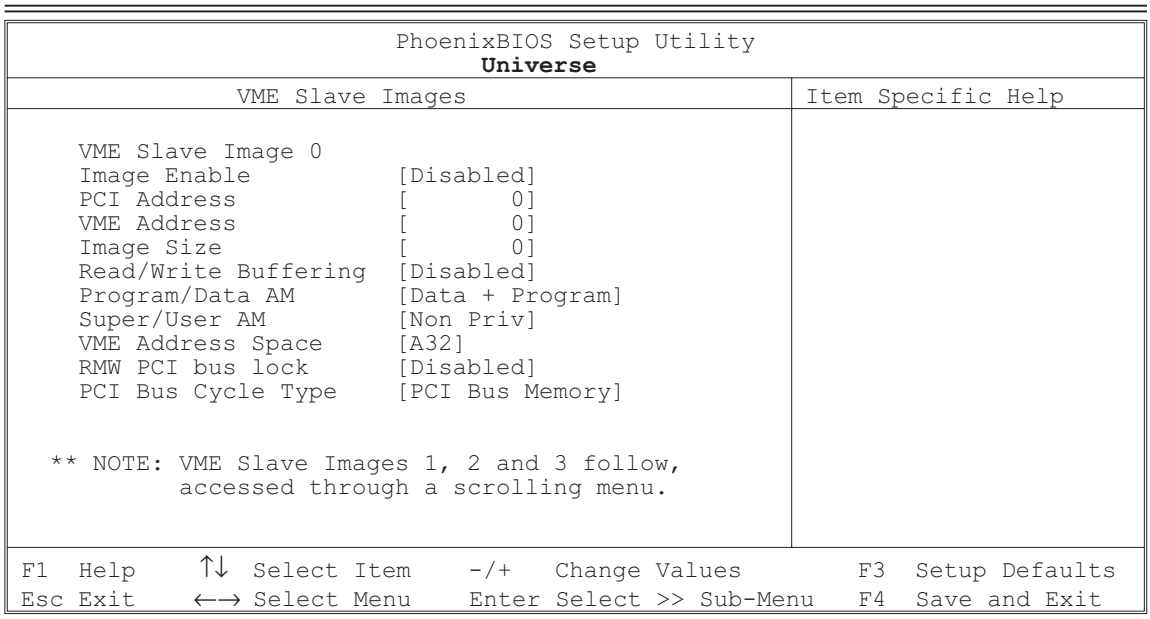


Figure 6-10 VME Slave Images Sub-Menu

Feature	Options	Description
Image Enable	Disabled Enabled	Enable or Disable this slave image.
PCI Address	(32-bit Address)	Hexadecimal PCI address for image. For image 0 this value is masked with 0xFFFFF000. For other images this value is masked with 0xFFFF0000.
VME Address	(32-bit Address)	Hexadecimal VME address for image. For image 0 this value is masked with 0xFFFFF000. For other images this value is masked with 0xFFFF0000.
Image Size	(32-bit Size)	Hexadecimal size of PCI/VME window to open in bytes. For image 0 this value is masked with 0xFFFFF000. For other images this value is masked with 0xFFFF0000.
Read/Write Buffering	Disabled Read Buffering Write Buffering Read and Write Buffering	Selects Buffering mode. Write buffering enables the Universe Posted Writes. Read buffering enables the Universe Prefetch Reads.
Program/Data AM	Data Program Data + Program	Selects Program or Data Address Modifier code to which the VME interface will respond.
Super/User AM	Non Priv Super	Selects Supervisor or Non Privileged Address Modifier code.
VME Address Space	A16 A24 A32 User 1 User 2	Selects VME bus address space.
RMW PCI Lock	Disabled Enabled	Enable/Disable of PCI bus locking for VME bus RMW cycle.
PCI Bus Cycle Type	PCI Bus Memory PCI Bus I/O PCI Bus config	Select PCI Bus Cycle type for transfer.

Table 6-10 VME Slave Images Options

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

6.2.9 Boot Setup Menu

This menu configures the boot strategy and the default boot devices.

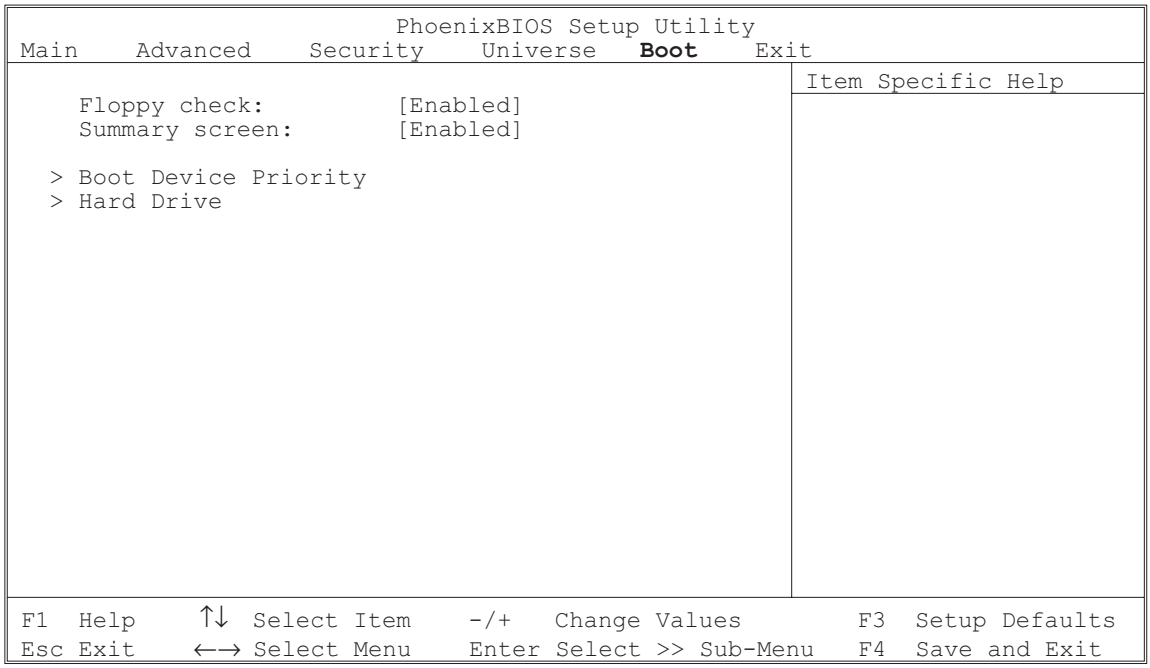


Figure 6-11 Boot Setup Menu

Feature	Options	Description
Floppy Check	Disabled Enabled	Verifies the floppy drive during boot. Disabling this option speeds up the boot process.
Summary Screen	Disabled Enabled	Specifies whether the configuration summary screen is displayed prior to boot.
Boot Device Priority	(see sub-menu)	Specifies the order of boot attempts.
Hard Drive	(see sub-menu)	Specifies the order of hard disk boot attempts.

Table 6-11 Boot Options

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

6.2.9.1 Boot: Boot Device Priority

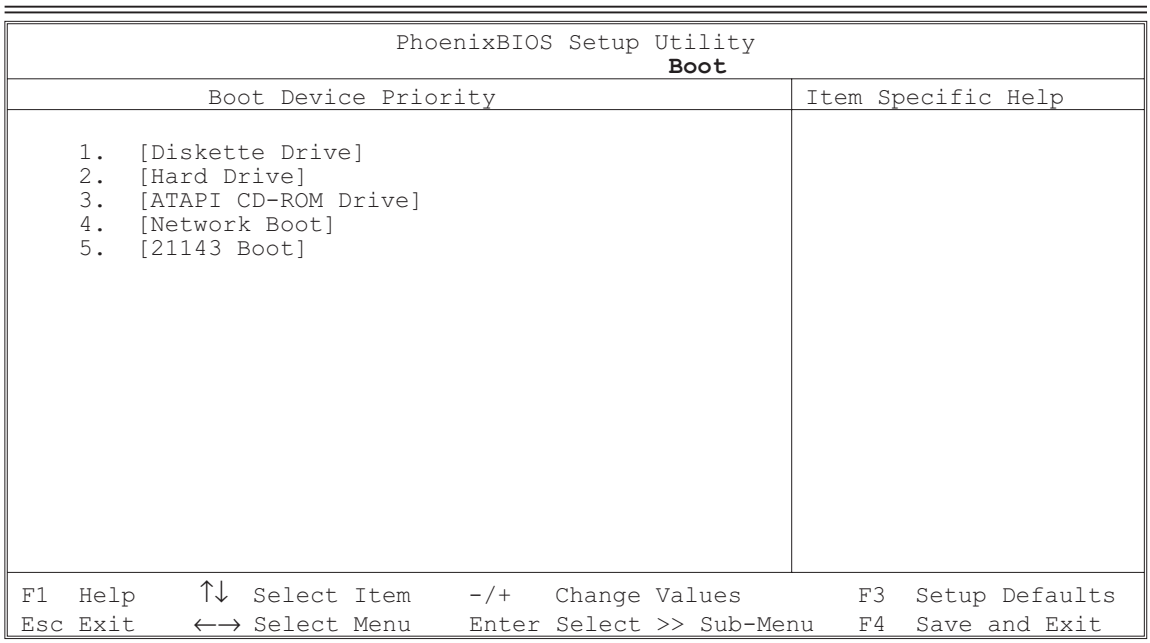


Figure 6-12 Boot Device Priority Sub-Menu

The boot order is changed by selecting a device, then changing its priority using '+' to move up the list (higher priority) and '-' to move down the list (lower priority).

The '21143 Boot' entry is the 'Etherboot' device as described in Section 6.7. The priority of this is fixed and will always be last.

The 'Network Boot' entry is for factory use only and should not be selected.

The Boot device can be manually selected from a menu at boot time by pressing the <Esc> key during the memory test.

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

6.2.9.2 Boot: Hard Drive

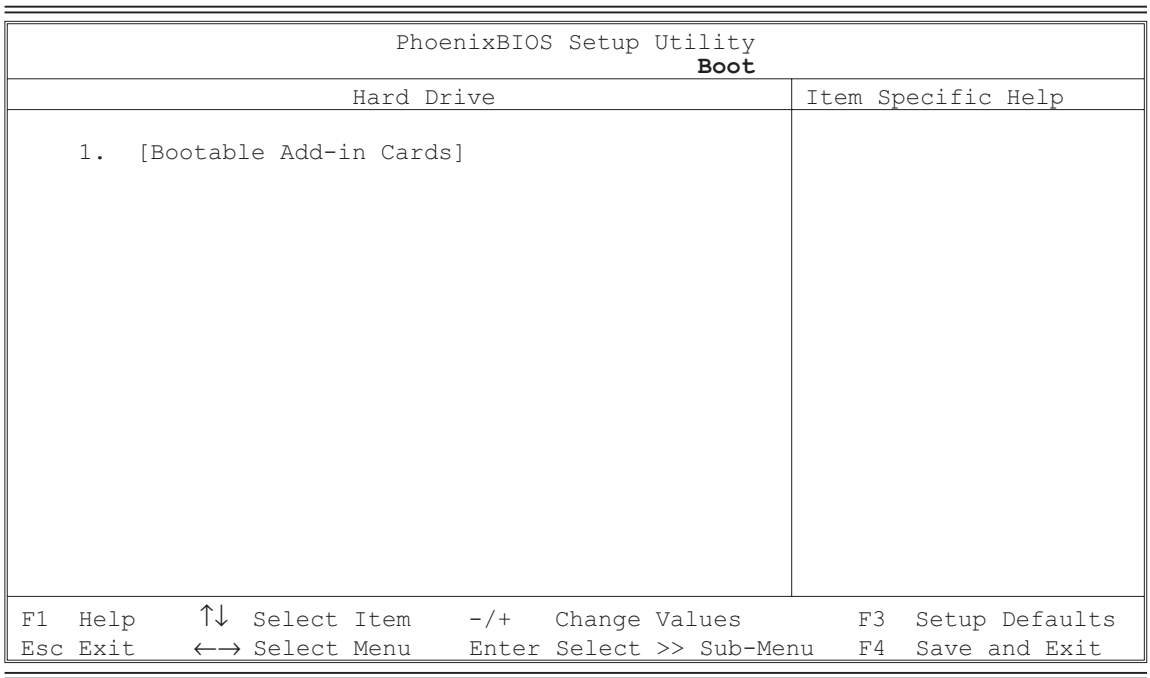


Figure 6-13 Hard Drive Sub-Menu

The boot order is changed by selecting a device, then changing its priority using '+' to move up the list (higher priority) and '-' to move down the list (lower priority).

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

6.2.10 Exit Menu

This menu provides options for saving or discarding the changes made to setup options, or the factory default values may be reloaded.

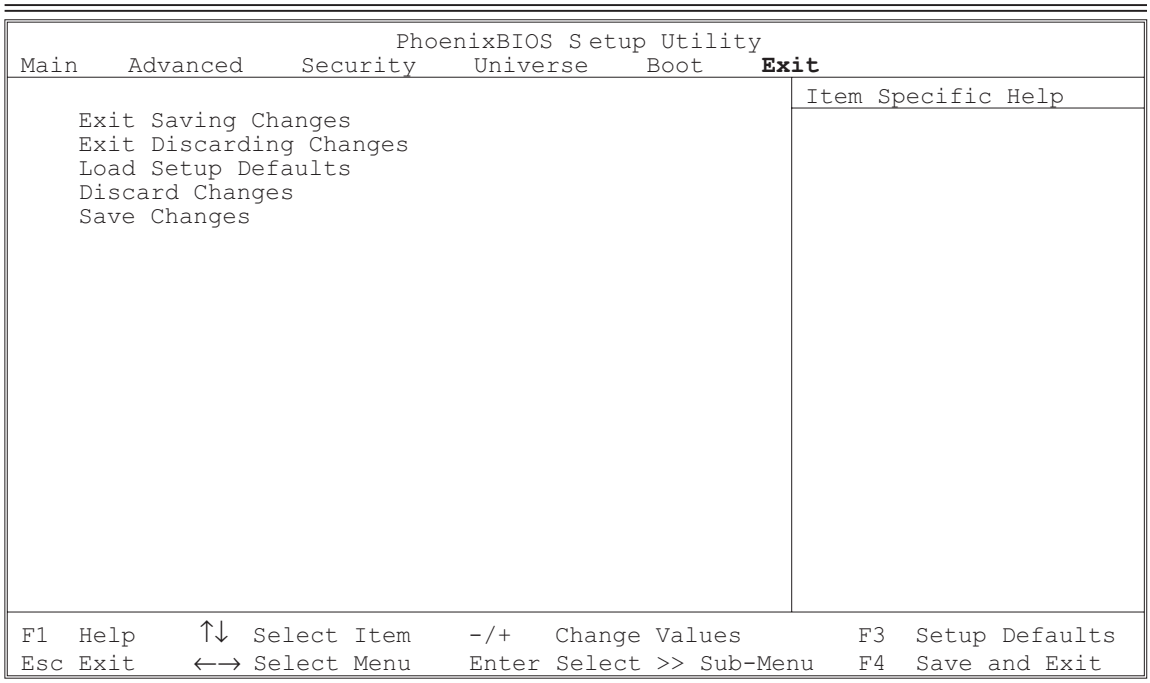


Figure 6-14 Exit Menu

Option	Description
Exit Saving Changes	Save setup changes to CMOS and Exit.
Exit Discarding Changes	Exit setup without saving changes.
Load Setup Defaults	Load factory defaults for all setup items.
Discard Changes	Reload setup data from CMOS, changes are lost.
Save Changes	Save modified setup data to CMOS, do not exit.

Table 6-12 Exit and Save Options

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

6.3 POST Errors

The VP PSE/P3x reports error conditions via the front panel POST LED. The errors are reported through combinations of long and short flashes – flash codes. Two types of error conditions are generated: recoverable and terminal.

Recoverable POST Errors

Whenever a recoverable error occurs during POST, PhoenixPICO BIOS displays an error message describing the problem.

PhoenixPICO BIOS also issues a flash code (one long flash followed by two short flashes) during POST if the video configuration fails (no card installed or faulty) or if an external ROM module does not properly checksum to zero.

An external ROM module (e.g., VGA) can also issue flash codes, usually consisting of one long flash followed by a series of short flashes.

Terminal POST Errors

There are several POST routines that issue a POST Terminal Error and shut down the system if they fail. Before shutting down the system, the terminal-error handler issues a flash code signifying the test point error, writes the error to port 80h, attempts to initialize the video, and writes the error in the upper left corner of the screen (using both mono and color adapters).

The routine derives the flash code from the test point error as follows:

1. The 8-bit error code is broken down to four 2-bit groups.
2. Each group is made one-based (1 through 4) by adding 1.
3. Short flashes are generated for the number in each group.

Example:

Testpoint 01Ah = 00 01 10 10 = 1-2-3-3 flashes

Test Points and Flash Codes

If the BIOS detects a terminal error condition, it halts POST after issuing a terminal error flash code and attempting to display the error code on upper left corner of the screen. It attempts repeatedly to write the error to the screen.

Code	Flashes	POST Routine Description
16	1-2-2-3	BIOS ROM checksum
20	1-3-1-1	Test DRAM refresh
22	1-3-1-3	Test 8742 Keyboard Controller
2C	1-3-4-1	RAM failure on address line xxxx*
2E	1-3-4-3	RAM failure on data bits xxxx* of low byte of memory bus
30	1-4-1-1	RAM failure on data bits xxxx* of high byte of memory bus
46	2-1-2-3	Check ROM copyright notice
58	2-2-3-1	Test for unexpected interrupts
98	1-2	Search for option ROMs. One long, two short flashes on checksum failure
B4	1	One short flash before boot

* Extended Error Codes

Table 6-13 POST Flash Codes

If the BIOS detects error 2C, 2E, or 30 (base 512K RAM error), it displays an additional word-bitmap (xxxx) indicating the address line or bits that failed. For example, “2C 0002" means address line 1 (bit one set) has failed. ”2E 1020" means data bits 12 and 5 (bits 12 and 5 set) have failed in the lower 16 bits.

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

6.4 Serial Console

Serial Console is a feature of the BIOS that redirects text-mode video output and keyboard input through the COM1 serial port. Serial Console is enabled by setting the CONSOLE MODE jumper (see Section 3.5 for jumper location and settings).

The default serial port parameters used by Serial Console are:

Baud Rate	9600
Data Bits	8
Stop Bits	1
Parity	None

The serial port baud rate can be changed via the **Serial Console Baud Rate** field of the BIOS setup utility. Since no flow control is used on the serial line, baud rates greater than 9600 require the use of a fast display terminal.

When Serial Console is enabled, all video BIOS INT10 text-mode character output is written to the serial port - including the BIOS initialization information and the BIOS setup screens. By connecting a VT100 terminal to COM1 all of the features of the BIOS setup utility can be accessed.

Serial Console uses a subset of the VT100 escape sequences to implement the BIOS setup screen and recognizes the following keyboard escape sequences for navigating the setup fields.

Key	Escape Sequence
Cursor Up	<ESC>[A
Cursor Down	<ESC>[B
Cursor Right	<ESC>[C
Cursor Left	<ESC>[D
PF1	<ESC>OP
PF2	<ESC>OQ
PF3	<ESC>OR
PF4	<ESC>OS

Serial Console display output is implemented by intercepting calls to INT 10h (video BIOS). The following video BIOS functions are implemented.

Function N°	Function
00H	Set Video Mode (Only sets internal variable)
02H	Set Cursor Position
09H	Write character and attribute at cursor
0AH	Write character at cursor
0EH	Write character in teletype mode
0FH	Get video mode (Read internal variable)

Serial Console works in parallel with any existing video and keyboard hardware. This means that on a board where Serial Console is enabled and a video card is installed, text will appear on both the video card's display and the VT100 terminal. The standard PC keyboard will also work in parallel with the VT100 terminal's keyboard.

Serial Console can only be used for standard text-mode DOS applications that do not write directly to the video memory.

NOTE Because the key-code ESC is used as the prefix to cursor and function keys, the user must press the ESC key twice to generate an ESC key-code in an application.

NOTE Because the video card and Serial Console work in parallel, discrepancies may be noticed when using boot disks created by Windows 95 and other operating systems. Text will appear on the video card display, but not all text will appear on the VT100 display terminal.

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

6.5 SCSI BIOS

The SCSI BIOS is the bootable ROM code that manages SCSI hardware resources. It is specific to the Symbios family of SCSI controllers or processors. This Symbios SCSI BIOS is integrated with the standard system PhoenixPICO BIOS, extending the standard disk service routine provided through INT13h.

During the boot time initialization, the SCSI BIOS determines if there are other hard disks, such as an EIDE drive, already installed via the system BIOS. If there are, the SCSI BIOS maps any SCSI drives it finds behind the drive(s) already installed. Otherwise, the SCSI BIOS installs drives starting with the system boot drive. In this case, the system boots from a drive controlled by the SCSI BIOS. This version of the Symbios BIOS supports the BIOS Boot Specification (BBS).

The next section, "Boot Initialization with BIOS Boot Specification (BBS)," discusses selecting boot and drive order.

6.5.1 Boot Initialization with BIOS Boot Specification (BBS)

The Symbios SCSI BIOS provides support for the BIOS Boot Specification (BBS), which allows you to choose which device to boot from by selecting the priority.

You will use the system PhoenixPICO BIOS setup menu to select the boot and drive order. In the system BIOS setup, the Boot Connection Devices menu appears with a list of available boot options. Use that menu to select the device and rearrange the order. Then exit to continue the boot process.

6.5.2 CDROM Boot Initialization

The Symbios SCSI BIOS supports boot initialization from a CDROM drive. There are five types of emulation:

- No emulation disk
- Floppy 1.2 MB emulation disk
- Floppy 1.44 MB emulation disk
- Floppy 2.88 MB emulation disk
- Hard disk emulation

The drive letter for the CDROM is assigned based on the type of emulation. For example, if a 1.44 MB floppy emulation CD was loaded, then the CDROM drive would become the designated A: drive, and the existing floppy would become drive B:.

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

6.6 Starting the SCSI BIOS Configuration Utility

This version of the Symbios SCSI BIOS includes the Symbios SCSI BIOS Configuration Utility. This Utility allows you to change the default configuration of your SCSI host adapters. You may decide to alter these default values if there is a conflict between device settings or if you need to optimize system performance.

You can see the version number of your SCSI BIOS in a banner displayed on your computer monitor during boot. If the utility is available, a message also appears on your monitor (for about five seconds) that looks like this:

```
Press Ctrl-C to start Symbios Configuration Utility...
```

This message remains on your screen for about five seconds, giving you time to start the utility. If you decide to press "Ctrl-C", the message changes to:

```
Please wait, invoking Symbios Configuration Utility...
```

After a brief pause, your computer monitor displays the Main Menu of the Symbios SCSI BIOS Configuration Utility.

To make changes with this menu driven utility, one or more Symbios SCSI host adapters must have NVRAM (non-volatile random access memory) to store the changes.

NOTE This NVRAM is managed by the system PhoenixPICO BIOS, and not all versions of this BIOS support the NVRAM functions. If NVRAM is not supported by your version of the system BIOS, configuration changes will be lost when the board is reset or powered off. Please contact Concurrent Technologies to obtain a BIOS firmware update if this feature is required in your application. The Main Menu will show if the system BIOS supports NVRAM.

CAUTION The SCSI BIOS Configuration Utility is a powerful tool. If, while using it, you somehow disable all of your controllers, pressing Ctrl-E after memory initialization during reboot allows you to re-enable and reconfigure.

NOTE Not all devices detected by the Configuration Utility can be controlled by the BIOS. Devices such as tape drives and scanners require that a device driver specific to that peripheral be loaded.

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

6.6.1 Main Menu

When you start the Symbios SCSI BIOS Configuration Utility, the Main Menu appears. This menu displays a list of up to four Symbios PCI to SCSI host adapters in your system and information about each of them.

To select an adapter, use only the arrow keys and enter key. Then you can view and/or change the current settings for that adapter, and the SCSI devices attached to it.

You can select an adapter only if Current Status is "On". Changes are possible only if NVRAM on your adapter is present. Selections that are not permissible will be grayed out.

The Main Menu looks like Figure 6-15.

```
Main Menu
               Port           Irq     --------Status---------  NvRAM
               Num            Level Current  Next-Boot         Found
SYM53C875      F800            11      On       Off             Yes
Change Adapter Status
Adapter Boot Order
Additional Adapter Configuration
Display Mode = Verbose
Mono/Color
Language
Help
Quit
```

Figure 6-15 SCSI BIOS Main Menu

Below the list of host adapters on the Main Menu display, you see eight options. They are described in detail below. If these settings are altered, the system reboots upon exit from the Configuration Utility via the Quit option.

6.6.2 Change Adapter Status

The change adapter status allows you to activate or deactivate a host adapter and all SCSI devices attached to it. When this option is used to make a change, the change takes place after a reboot that is automatic upon exit from the utility. The Change Status on Next Boot menu looks like Figure 6-16.

```
Main Menu
Change Status on Next Boot:
               PortIrq                 --------Status---------  NvRAM
               Num            Level Current  Next-Boot         Found
SYM53C875      F800           11      On       On              Yes
```

Figure 6-16 SCSI BIOS Change Adapter Status Menu

To change an adapter's status, select it and press Enter. Then press the Escape (Esc) key to exit from this menu.

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

6.6.3 Adapter Boot Order

The adapter boot order allows you to set the order in which host adapters will boot when you have more than one Symbios host adapter in your system. When this option is selected, the Boot Order menu appears, as shown in Figure 6-17.

```
Main Menu

BootSeq           Bus    DevFunc
0     SYM53C875   00     A0
```

Figure 6-17 SCSI BIOS Adapter Boot Order Menu

To change an adapter's boot order, select it and press Enter. You are then prompted to enter the new boot sequence number. To remove an adapter's boot order, press Enter again rather than entering a new sequence number. While the maximum capacity is 32 adapters, only 0 through 3 can be assigned a boot order. If an invalid number is entered, an error message appears. When the adapters are ordered as desired, press the Escape (Esc) key to exit from this menu and reboot.

6.6.4 Additional Adapter Configuration

The additional adapter configuration allows you to configure an adapter that is not assigned a boot order. When this option is selected, the Adapter Configuration menu appears, as shown in Figure 6-18.

```
Main Menu
BootSeq           Bus    DevFunc
1     SYM53C875   00     A0
```

Figure 6-18 SCSI BIOS Additional Adapter Configuration Menu

Highlight the adapter to be configured and press Enter. The message "Resetting Adapter, Please wait" appears, and then the system scans for devices. Finally, the Utilities Menu appears and lists the available options, which are described below.

6.6.5 Display Mode

The Display Mode option determines how much information about your host adapters and SCSI devices appear on your computer monitor during boot. For more complete information, choose the verbose setting. For a faster boot, choose the terse setting.

6.6.6 Mono/Color

The Mono/Color option allows a choice between a monochrome or color display for the SCSI BIOS Configuration Utility. You might need to choose the mono setting to get a more readable screen on a monochrome monitor.

6.6.7 Language

If enabled, the Language option allows you to select from five languages for the Configuration Utility: English, German, French, Italian, and Spanish.

6.6.8 Help

The Help option brings up a help screen with information about the Main Menu.

6.6.9 Quit

The Quit option allows exiting from the SCSI BIOS Configuration Utility when the Main Menu is displayed.

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

6.6.10 Esc

The Esc option allows exiting from all the screens except the Main Menu. Press the Esc key to use this option.

6.6.11 Utilities Menu

When you select a host adapter on the Main menu, the Utilities menu appears, as shown in Figure 6-19.

```
SYM53C875
     Utilities

Adapter Setup
Device Selections
Help
Exit this menu
```

Figure 6-19 SCSI BIOS Utilities Menu

Choose Adapter Setup to view and change the selected adapter settings. Choose Device Selections to view and change settings for the devices attached to the selected adapter.

You are returned to this menu after making changes to the configuration of any host adapter or connected SCSI device. Before you exit this menu, you are prompted to save or cancel any changes.

6.6.11.1 Adapter Setup Menu

When you select Adapter Setup, the Adapter Setup menu appears, as shown in Figure 6-20.

```
SYM53C875
    Adapter

Setup SCAM Support                  Off
Parity                              Enabled
Host SCSI ID                        7
Scan Order                          Low to High <0..Max>
Removable Media Support             None
CHS Mapping                         SCSI Plug & Play Mapping
Spinup Delay(secs)                  2
Help
Restore Default Setup
Exit this menu
```

Figure 6-20 SCSI BIOS Adapter Setup Menu

The settings in this menu are global settings that affect the selected host adapter and all SCSI devices attached to it. One of these choices can be selected by highlighting it and pressing Enter.

6.6.11.1.1 SCAM Support

The Symbios BIOS version 4.X supports the SCSI Plug and Play protocol called SCAM (SCSI Configured AutoMatically). SCAM support by default is off in versions 4.09 and later. You may choose to turn this on. Note that if this BIOS is flashed onto a board with existing settings, then these settings will not be changed to reflect the new BIOS defaults.

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

6.6.11.1.2 Parity

The Symbios PCI to SCSI host adapters always generate parity, but some older SCSI devices do not. Therefore, you are offered the option of disabling parity checking.

NOTE When disabling parity checking, it is also necessary to disable disconnects for all devices, as parity checking for the reselection phase is not disabled. If a device does not generate parity, and it disconnects, the I/O never completes because the reselection never completes.

6.6.11.1.3 Host SCSI ID

In general, it is suggested that you not change your host adapter ID from the default value of 7, as this gives it the highest priority on the SCSI bus.

6.6.11.1.4 Scan Order

This option allows you to tell the SCSI BIOS and your device drivers to scan the SCSI bus from low to high (0 to max) SCSI ID, or from high to low (max to 0) SCSI ID. If you have more than one device on the SCSI bus, changing the scan order changes the order in which drive letters are assigned by the system. Drive order may be reassigned differently in systems supporting the BIOS Boot Specification (BBS).

See the section "Boot Initialization for BIOS Boot Specification (BBS)" for more information.

NOTE This Scan Order option may conflict with operating systems that automatically assign a drive order.

6.6.11.1.5 Removable Media Support

This option defines the removable media support for a specific drive. When this option is selected, a window appears with three choices:

- None
- Boot Drive Only
- With Media Installed

None indicates there is no removable media support whether the drive is selected in BBS as being first, or first in scan order in non-BBS.

Boot Drive Only provides removable media support for a removable hard drive if it is first in the scan order.

With Media Installed provides removable media support wherever the drive(s) actually resides.

One of these choices can be selected by highlighting it and pressing Enter.

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

6.6.11.1.6 CHS Mapping

This option defines the cylinder head sector (CHS) values that will be mapped onto a disk without pre-existing partition information. SCSI Plug and Play Mapping is the default value.

To support interchange with non-compatible systems, there is another option that can be selected by choosing CHS Mapping and then cursoring to "Alternate CHS Mapping".

NOTE Neither of these options will have any affect after the disk has been partitioned with the FDISK command.

To remove partitioning, two options are available:

Reformat the disk using the Format Device option. See the section "Device Selections Menu" below.

Use the FDISK /MBR command at the C:\ prompt, where MBR represents master boot record.

Important: Reformatting the disk or using FDISK /MBR erases all partitioning and data that exists. Be careful when using either the Format utility or the FDISK /MBR command that you target the correct disk.

NOTE After clearing the partitioned data, it is necessary to reboot and clear memory or the old partitioning data will be reused, thus nullifying the previous operation.

6.6.11.1.7 Spinup Delay (secs)

This option allows you to stagger spin ups for a longer period of time to balance the total current load. The default value is 2 seconds with choices between 1 and 10 seconds.

This is a power management device designed to accommodate disk devices that may have heavy current load during power up. If multiple drives are being powered up simultaneously and drawing heavy current loads, then this option staggers the spin ups to limit startup current.

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

6.6.11.2 Device Selections Menu

When you select the Device Selections option, the corresponding menu appears, as shown in Figure 6-21.

```
SYM53C875
Device Selections 0-7

                  Sync  Data  Disc Time    Scan        Queue   Initial
                  Rate  Width      Out   Bus   LUNS    Tags    Boot

Seagate 31051N    20     8    On    10    Yes    Yes   On      No
1-Dev1 N/A        80     8    On    10    Yes    Yes   On      No
2-Dev2 N/A        80     8    On    10    Yes    Yes   On      No
Quantum Lightning 20     8    On    10    Yes    Yes   On      No
4-Dev4 N/A        80     8    On    10    Yes    Yes   On      No
IOMega ZIP 100    20     8    On    10    Yes    Yes   On      No
6-Dev6 N/A        80     8    On    10    Yes    Yes   On      No
Device Selections   8-15
Help
Exit this menu
```

Figure 6-21 SCSI BIOS Device Selections Menu

The settings in this menu affect individual SCSI devices attached to the selected host adapter. Changes made from this menu do not cause the system to reboot upon exit from the SCSI BIOS Configuration Utility. To change a value, select the required device by using the arrow keys and press Enter. A new menu appears providing the options and utilities available. For example, you could cursor to Sync Rate to change the Sync Rate value of the chosen device.

6.6.11.2.1 Device Name

```
Inside device
          Sync Rate
          Width
          Disconnect
          Read/Write I/O Timeout (secs)
          Scan for Device at Boot Time
          Scan for SCSI LUNs
          Queue Tags
          Initial Boot
          Format
          Verify
          Help
          Restore Default Setup
            Exit this menu
```

Figure 6-22 Device Options Menu

Please review the descriptions of each option below before changing any values.

6.6.11.2.1.1 Sync Rate (mega Bytes/sec)

This option defines the maximum data transfer rate the host adapter will attempt to negotiate. The host adapter and a SCSI device must agree to a rate they can both handle.

6.6.11.2.1.2 Width (bits)

This option defines the maximum SCSI data width the host adapter will attempt to negotiate. The host adapter and a SCSI device must agree to a width they can both handle. Only host adapters that can do 16-bit data transfers have this option enabled.

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

6.6.11.2.1.3 Disconnect

SCSI devices have the ability to disconnect from the initiator during an I/O transfer. This option frees the SCSI Bus to allow other I/O processes. This option tells the host adapter whether or not to allow a device to disconnect. Some devices run faster with disconnects enabled (mostly newer devices), while some run faster with disconnects disabled (mostly older devices).

6.6.11.2.1.4 Read Write I/O Timeout (secs)

This option sets the amount of time the host adapter waits for a read, write, or seek command to complete before trying the I/O transfer again. Since this provides a safeguard allowing the system to recover if an I/O operation fails, it is recommended that you always set the time-out to a value greater than zero. A zero value allows unlimited time for an operation to complete and could result in a hang if the operation couldn't complete.

6.6.11.2.1.5 Scan for Device at Boot Time

Set this option to "No" if there is a device that you do not want to be available to the system. Also, on a bus with only a few devices attached, you can speed up boot time by changing this setting to "No" for all unused SCSI Ids.

6.6.11.2.1.6 Scan for SCSI Logical Units (LUNs)

Set this option to "No" if you have problems with a device that responds to all LUNs whether they are occupied or not. For example, if a SCSI device with multiple LUNs exists on your system but you do not want all of those LUNs to be available to the system, then set this option to "No." This will limit the scan to LUN 0 only.

6.6.11.2.1.7 Queue Tags

This option enables or disables the issuing of queue tags during I/O requests when your device driver can do this.

6.6.11.2.1.8 Initial Boot

This option allows any device attached to the first adapter to become the boot device. It provides the users of non-BBS personal computers some of the flexibility of a BBS machine.

6.6.11.2.1.9 Format

If enabled, this option allows low-level formatting on a magnetic disk drive. Low-level formatting will completely and irreversibly erase all data on the drive.

NOTE Formatting will default the drive to a 512 byte sector size even if the drive had previously been formatted to another sector size.

6.6.11.2.1.10 Verify

This option reads all the sectors on a disk to check for any errors. When selected, this option displays the following message:

Verify all sectors on the device
Press ESC to abort
Else press any key to continue

6.6.11.2.1.11 Help

This option provides a help screen with information about the current menu.

6.6.11.2.1.12 Restore Default Setup

This option resets all device selections back to their optimal settings. Select this option to restore all manufacturing defaults for the specified adapter. Note that all user customized options will be lost upon saving after restoring default setup.

6.6.11.2.1.13 Exit this Menu

This option leaves the current menu screen and returns to the previous screen.

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

6.6.12 To Exit the SCSI BIOS Configuration Utility

Since some changes only take effect after your system reboots, it is important that you exit this configuration utility properly. Return to the Main menu and exit via the Quit option. If you reboot the system without properly exiting from this utility, some changes may not take effect.

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

6.7 Network Boot ROM

The BIOS firmware includes a BIOS extension for the Intel 21143 Ethernet controller which allows remote booting of some operating systems.

The BIOS extension is built from the “Etherboot” software and is pre-configured to use the “bootp” and TFTP protocols.

See http://etherboot.sourceforge.net for more information on “Etherboot” and remote booting.

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

VME System Architecture Test Handler

7.1 Introduction

The VME System Architecture (VSA) Test Handler firmware provides an environment where interactive testing may be performed on one or more Concurrent Technologies' VME CPU boards.

The level of testing provided by VSA is more comprehensive than that provided by the BIOS POST, and testing can also be looped to aid diagnosis of intermittent faults. Failing tests provide diagnostic information that can be used to identify the cause of the problem.

VSA also provides a number of utility BISTs through which memory and I/O may be examined or modified. PCI devices can also be identified and their configuration registers displayed and changed.

VSA allows all Concurrent Technologies' boards in a system to be tested from a single console connected to the System Controller. This console can be a standard VGA PMC and keyboard, or a serial terminal connected to COM1.

7.2 The VSA Environment

7.2.1 Slot Numbering

Throughout the VSA firmware, boards are identified by their logical slot number. This number does not represent the physical backplane slot. The logical slot number is assigned by the Master Test Handler when it detects a board capable of participating in system testing. Therefore boards from other manufacturers, boards not jumpered for VSA mode and boards that are seriously damaged will not be detected and included in the numbering scheme.

In a system containing a faulty board, which is unable to enter its Slave Test Handler, the logical slot numbers will not match the board positions in the rack. The faulty board can be isolated by executing a short BIST (e.g. BIST 126) on each board in turn. The USER LED is illuminated while the BIST is executing allowing the faulty board to be quickly identified.

7.2.2 VSA Console Devices

VSA can use the VGA compatible display interface and keyboard for the console device, or a serial terminal may be used connected to COM1. VSA automatically detects the users choice of console when the attention, "U" keystroke is entered at the prompt.

For serial consoles VSA will auto-detect the baud rate, though it is recommended that 19,200 Baud or 9,600 Baud be used. The auto-baud function may require the attention key to be pressed more than once in noisy environments, or when lower rates than the recommended ones are used.

Console I/O will only be provided by the Master Test Handler.

7.2.3 Starting the Master Test Handler

VSA mode is selected by fitting the "VSA" jumper as indicated in Section 3.6. The board will enter VSA mode before the BIOS starts displaying sign-on text, so the first console output will be the VSA user attention prompt.

When VSA starts, it outputs attention characters to all possible console devices simultaneously; i.e. the default video adapter and both COM ports. On the video adapter this will appear as a series of asterisks "*", on a serial console the appearance of the character will depend on the console's baud rate.

Only the board in the system controller slot can provide console output. This is the Master Test Handler (MTH); if other boards are present they will automatically enter their Slave Test Handlers (STH) and await BIST execution requests from the Master.

When the attention prompt is displayed, pressing "U" (capital "u") on the desired console device will display an option menu. From here the Master Test Handler can be entered, or the board booted back to BIOS mode.

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

7.2.4 Remote Testing from the System Controller

In a system comprising more than one Concurrent Technologies' CPU board, only the system controller board will provide a console interface; however, this board can be used to test the other VSA jumpered boards through their Slave Test Handlers.

During system startup, the VSA jumpered system controller will detect all other functional, VSA jumpered boards. These boards will be identified via the VSA startup screen, together with their logical slot ID.

The default test slot is initially set as slot zero (the system controller). This can be changed using the SLOT command to any valid logical slot. When the default slot is changed, all tests will run on the slave board and the TESTMENU command will return the list of valid BISTs for that board.

In a system containing a faulty board, which is unable to enter its Slave Test Handler, the logical slot numbers will not match the board positions in the rack. The faulty board can be isolated by executing a short BIST (e.g. BIST 126) on each board in turn. The USER LED is illuminated while the BIST is executing allowing the faulty board to be quickly identified.

7.2.5 Bootloading the BIOS

The board can be booted to BIOS mode from VSA mode using the BPHASE command. Once in BIOS mode, warm boots will confine execution to the BIOS firmware; however, if a cold boot is generated without changing the VSA jumper, the board will re-enter VSA mode.

If the system controller board is booted to VSA mode and the "U" command is not entered, the firmware will proceed to boot the board back to BIOS mode.

7.2.6 BIST Execution

BIST execution is started using the TEST command. While a test is executing, no further commands may be entered.

It is possible to specify more than one BIST for execution using the ";" separator, e.g.

`T14;T15;T20,4` Execute Test 14, Test 15 and Test 20. Test 20 has a command parameter.

BISTs may be executed more than once, automatically, using the iteration count. Using an iteration count of zero will execute the BIST until the break "Ctrl-C" command is pressed, e.g.

`10<T14>` Execute T14 ten times.

`0<T5>` Execute T5 "forever".

The iteration counter can also be used to execute a sequence of tests more than once, e.g.

`5<T14;T15;T16>` Execute T14, T15 and T16 five times.

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

7.3 MTH Command Reference

This section details all of the commands available from the MTH (Master Test Handler) prompt. Commands are divided into General and Utility sections. The list below shows the commands in uppercase letters only, but lowercase letters may also be used.

Where numbers are entered decimal notation is assumed unless the value ends in 'H' or 'h' In this case the value is assumed to use hexadecimal notation.

7.3.1 Help Screens

Typing HELP at the MTH prompt will give general help for the MTH commands. Help for the utility commands is available with the UTILHELP command.

7.3.2 General Commands

The general commands control BIST execution on the local or a remote test slot.

BPHASE [Short command **B**]

Boot the BIOS firmware on the default slot.

BPHASE # [Short command **B**]
- slot number

Boot the BIOS firmware on the specified slot.

CLEARSUM [Short command **C**]

Clears the pass and fail counts for all BISTs on the default slot.

ID [No short command]

Displays a list of boards in the system, their status and the active default slot number. Boards are identified by logical slot number. Only Concurrent Technologies boards, jumpered for VSA mode will be identified by this command.

PRINT [Short command **PR**]

Toggles the BIST printing flag. When PRINT is off BIST diagnostic messages are not displayed during testing, only the pass or fail and error code are displayed.

RESET [Short command **R**]

This command can only be applied to a slave board, which can be made to reset and reboot into its Slave Test Handler.

SLOT [Short command **S**]

Displays the default test slot, i.e. the slot on which BISTs will execute. The slot number does not relate to the physical backplane slot, it is a logical slot number assigned by the test handler.

SLOT # [Short command **S**]
- slot number

Changes the default slot number. The slot number does not relate to the physical backplane slot, it is a logical slot number assigned by the test handler.

It is possible to specify an unoccupied slot number with this command; however, a warning message will be displayed and confirmation of the change requested.

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

SUM [No short command]

Prints the pass and fail counts for all BISTs available on the default slot.

SUM # [No short command]
- test number, in the range 0-255

Prints the pass and fail count, for the BIST indicated, on the default slot.

TEST # [Short command **T**]

- test number, in the range 0-255

Starts BIST execution on the default slot. The test is run without parameters. Further command input is prevented until the BIST completes (or a BIST time-out is generated for a remote slot).

TEST #,p1,p2,… [Short command **T**]
- test number, in the range 0-255
p1,p2 - test parameters (see individual BIST descriptions for details)

Starts test execution on the default slot, the supplied parameters are passed to the BIST

TESTMENU [Short command **TM**]

Displays a list of available BISTs for the default slot, together with their associated test number.

UTILHELP [Short command **U**]

Displays the help screen for the utility commands, described below.

VERSION [Short command **V**]

Prints the firmware version number on the default slot.

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

7.3.3 Utility Commands

IRO, IRR, ICR, ICW

These commands are reserved for factory testing. They report and modify the state of the VSA board communication data-structures.

INB port_address	read a byte from the specified I/O address
INW port_address	read a word from the specified I/O address
IND port_address	read a dword from the specified I/O address
OUTB port_address,data	write a byte to the specified I/O address
OUTW port_address,data	write a word to the specified I/O address
OUTD port_address,data	write a dword to the specified I/O address
DB address,length	read a byte from the specified memory address
DW address,length	read a word from the specified memory address
DD address,length	read a dword from the specified memory address
DQ address,length	read a qword (64-bits) from the specified memory address
SB address,data	write a byte to the specified memory address
SW address,data	write a word to the specified memory address
SD address,data	write a dword to the specified memory address

NOTE The I/O and memory read and write functions only operate on the local slot, i.e. the Test Master. To read or write I/O and memory on the slave boards: change the default slot number and use the equivalent BIST function TEST 101-104.

TEA	toggle Test Error Action flag between QUIT and CONTINUE
SEA	toggle Sequence Error Action flag between QUIT and CONTINUE

The last two commands prevent looped BIST execution from halting if an error occurs.

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

This page has been intentionally left blank

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

VSA Mode Diagnostics

This chapter describes the board's initialization into VSA mode and the Tests that can be run from VSA mode. For details of the VSA command line interface, refer to chapter 7.

For a description of the board's operation in BIOS mode, refer to chapter 6 of this manual.

8.1 Initialization Checks

The board will always start executing PC BIOS firmware; however, if the VSA Mode jumper is fitted the BIOS will transfer control to the VSA firmware once it has completed chipset initialization, cache and memory sizing.

The VSA firmware performs additional hardware initialization and some basic functional checks before switching to Protected Mode and entering its master or slave test handler. These functional checks are described below.

8.1.1 Check 16: CPU Alive Check

To test the basic CPU-interconnect access path, the CPU writes the ID of this test to the BIST TEST ID Interconnect register, then reads it back to verify that it was correctly written.

The test fails if the value read is not the same as the value written.

8.1.2 Check 18: Scratchpad RAM Check

The first 192 Kbytes of RAM, the scratchpad, are used by the BIST firmware. This memory area is tested by writing and verifying two rotating test patterns across the scratchpad address range. The first pattern is 0AA55h, the second 055AAh. Each pattern is rotated left two bit positions for each increment of the address; this ensures that consecutive addresses have unique data patterns whether they use 16-, 32- or 64-bit bus fetches.

NOTE This is the only test carried out on this area of RAM - all other BISTs test only the remaining RAM area.

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

8.2 BIST Descriptions

The following is a list of the tests that are available in the firmware set installed on this board, together with an overview of the function of each test.

A description of each possible error condition, with its code, is given for each test.

8.2.1 Test 1: Test Initialization Routine

This pseudo-test performs no actual testing of the board. It sets up in RAM several data values, such as RAM size, that are used by later tests. This BIST should always be run at the start of a test session before any other tests are run.

This test will be executed at power-up, and may be invoked thereafter by a Master Test Handler.

8.2.2 Test 2: PROM Check

This BIST performs a checksum test over the VSA firmware installed in the BIOS EPROM. By default, the range tested is from 0FFF82100h to 0FFFC6000h.

The test range is configurable by the user of the board: the parameters that control the test are stored in three consecutive 32-bit words at the start of the VSA firmware, i.e. starting at address 0FFF82010h. The parameters are as follows:

0FFF82010h: Checksum Area Start Address
0FFF82014h: Checksum Area Length (in bytes)
0FFF82018h: Expected Checksum Value

A feature of the test is that if the expected checksum value is set to a value of 0FFFFFFFh (-1 in decimal) then the test will always pass, but will report the actual checksum value to the test master. This is useful for discovering the new checksum value of a modified range.

Note that if the checksum area is defined to cover the three words that control the test, it will not be possible to calculate an expected checksum value.

Error codes:

0300h: The checksum test failed.

8.2.3 Test 4: Numeric Coprocessor Test

This BIST performs checks on the functions of the numeric coprocessor component of the CPU.

At the start of the test, the coprocessor is re-initialized using an FINIT instruction and the required operating mode set up. Basic arithmetic functions are checked and a deliberate division by zero is attempted in order to generate an exception condition and the associated interrupt.

If the results of the arithmetic operations are incorrect or result in an exception, or if no divide by zero exception is generated the test fails.

The exception test is repeated with the external FERR signal enabled in place of the internal one.

Error codes:

04x1h - Error in re-initializing floating-point processor

04x2h - Computation generated an exception or incorrect result
04x3h - Exception occurred in floating-point comparison
04x4h - Divide by zero failed to generate the correct exception status

x - 0 For CPU internal interrupt enabled.

1 For external FERR# interrupt enabled.

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

8.2.4 Test 6: Interconnect Image Check

This BIST reads and verifies the vendor ID and the board name from the Header Record of the local Interconnect Template. The interconnect template is a data structure used by VSA to communicate between boards.

Error codes:

0300h - Image check failed.

8.2.5 Test 7: Off-board Interconnect Access

This BIST searches for a known interconnect record in the interconnect template of the System Controller board. The interconnect template is a data structure used by VSA to communicate between boards.

Error codes:

0300 - Test failed

8.2.6 Test 9: 8254 PIT Test

This BIST checks the PC compatible, Programmable Interval Timers within the PIIX4. To test the secondary PIT (PIT2) see Test 40.

Each timer in turn is initialized with a start count value, then monitored to make sure that it counts successfully.

Error codes:

0401h - Timer 0 failed to count
0402h - Timer 1 failed to count
0403h - Timer 2 failed to count

8.2.7 Test 10: 8259A PIC Test

This BIST checks the functionality of the PC compatible Programmable Interrupt Controllers on the board.

Error codes:

0402h - Interrupt did not occur,
0412h - Incorrect interrupt occurred.

8.2.8 Test 12: Local RAM Fixed Pattern Test

This BIST performs a short test on local RAM. The range of memory to be tested is determined from data set up by the Test Initialization BIST. However, if the test is being run by a Master Test Handler, then calls are made using the BIST protocol which allow range override by the Master.

The Master can specify any memory range for the test; however, the video memory hole between A0000h and BFFFFh must be avoided.

First, the memory under test is initialized to 00000000. Then two “marches” are made through memory with patterns as follows:

pass 1 new=FFFFFFFF old=00000000

pass 2 new=00000000 old=FFFFFFFF

During the march through the memory range, for every 32-bit word location, first the old pattern is verified, then the new pattern is written and verified.

To reduce execution time when testing large regions of memory this BIST runs from DRAM; however, the area of memory from which the test code executes is first tested by the ROM-based version of the routine.

When this test is executed as a power-up BIST, it is necessary to limit execution time. Therefor, the power-up BIST will only test the area of RAM before the video memory hole; i.e. the power up BIST will test 30000H to 9FFFFH.

Error codes:

0300h - Test failed. For details see accompanying message.

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

8.2.9 Test 13: SCC Access

This BIST performs a read-after-write test on each serial channel of the board.

The serial controller's scratch register is used for this test. A write-then-read of a shifting one value is used to test access to the device. The value is written then read and verified.

Error codes:

04x0h - Failure on channel x. Accompanying message gives further details.

8.2.10 Test 20: Universe NMI Test

This BIST checks the ability of the universe to generate a NMI to the processor using the software generated interrupt via LINT1.

Error codes:

0406h - no interrupt generated or spurious interrupt

8.2.11 Test 23: Local RAM Read/Write Test

This BIST is a simple non-destructive read-complement-write test. The range of memory to be tested is determined from data set up by the Test Initialization BIST. However, if the test is being run by a Master Test Handler, then calls are made using the BIST protocol which allow range override by the Master.

The Master can specify any memory range for the test; however, the video memory hole between A0000h and BFFFFh must be avoided.

The test operates on double words throughout the range selected.

During the test, PCI Bus Error interrupts are enabled. If one of these should occur the test is aborted and a diagnostic message displayed.

To reduce execution time when testing large regions of memory this BIST runs from DRAM; however, the area of memory from which the test code executes is first tested by the ROM-based version of the routine.

Error codes:

0300h - Test failed. For details see accompanying message.
0402h - PCI bus error occurred.
0403h - PSB Error occurred.

8.2.12 Test 25: Local RAM Dual Address Test

This BIST checks for Dual Addressing in the RAM. The range of memory to be tested is determined from data set up by the Test Initialization BIST. However, if the test is being run by a Master Test Handler, then calls are made using the BIST protocol which allow range override by the Master.

The Master can specify any memory range for the test; however, the video memory hole between A0000h and BFFFFh must be avoided.

The BIST proceeds to write the memory address, rotated two bit positions, to each Dword location. When the whole test region has been written, the memory is read back and compared against the expected value. By using the memory address as test data, any incorrect values will identify the dual-addressed memory location. During the test, the PCI Bus Error interrupts are enabled. If one of these should occur, the test is aborted and a diagnostic message displayed.

To reduce execution time when testing large regions of memory this BIST runs from DRAM; however, the area of memory from which the test code executes is first tested by the ROM-based version of the routine.

Error codes:

0300h - Test failed; associated message gives details.
0402h - PCI bus error occurred.
0403h - PSB Error occurred.

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

8.2.13 Test 27: Local RAM Execution Test

This BIST executes code from RAM in the selected test region. The range of memory to be tested is determined from data set up by the Test Initialization BIST. However, if the test is being run by a Master Test Handler, then calls are made using the BIST protocol which allow range override by the Master.

The Master can specify any memory range for the test; however, the video memory hole between A0000h and BFFFFh must be avoided.

This test copies a small string of code into the selected RAM area and executes out of that RAM. The buffer is first filled with INT3 opcodes, and then the sequence of instructions is copied to the beginning of the buffer. A jump is made to the code, which copies itself to the next available location in the buffer, then overwrites the old copy with INT3 instructions once more. If an error occurs such that it the code jumps into a location outside the instruction sequence, this is trapped via the INT3 instructions.

When the test code reaches the end of the buffer, it returns to the caller and the test has passed.

During the test, the PCI Bus Error interrupts are enabled. If one of these should occur, the test is aborted and a diagnostic message displayed.

Error codes:

0300h - Test failed: for details see accompanying message.
0402h - PCI bus error occurred.
0403h - PSB Error occurred.

8.2.14 Test 28: SCC Interrupt Test

This BIST checks that each serial channel on the board is capable of generating an interrupt.

A null character is transmitted on each channel in turn to generate a transmit interrupt from that channel. If the interrupt occurs, checks are made to ensure that there is a transmit interrupt pending on the serial device. A channel specific interrupt is generated for each channel in turn.

Error codes:

The error number (04xyh) is encoded in the following way:

x - Channel Number for which interrupt was generated.

y - 1 No interrupt,

2 Wrong interrupt,

3 No interrupt pending indicated,

4 No TX interrupt pending.

8.2.15 Test 29: SCC Internal Loopback Test

This BIST performs an Internal Loopback Test on each serial channel of the board.

Each channel in turn is switched into Internal Loopback Mode, and 255 characters are transmitted and received. The data received is checked against the data sent (ascending byte values, 0..254). The test runs in asynchronous mode at 9600 baud.

Error codes:

04x1h - timed out waiting for TxRDY,
04x2h - timed out waiting for RxRDY,
04x3h - data mismatch on compare after write.

where x is the Channel.

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

8.2.16 Test 30: SCC External Loopback Test

This BIST performs an External Loopback Test on the serial channel of the board.

The serial channel should be looped back externally by connecting TX0 to RX0, DSR0 and RI0 to DTR0, RTS0 and CTS0 to CD0.

The test sequence is identical to the Internal Loopback Test described above, followed by a test in which the modem signals are manipulated independently to check their functionality.

Error codes:

The error number (04xyh) is encoded in the following way:

x = channel under test,

y = error type.

1. timed out waiting for TxRDY,
2. timed out waiting for RxRDY,
3. data mismatch on compare after write,
4. RTS Active, message gives details,
5. RTS Inactive, message gives details,
6. DTR Active, message gives details,
7. DTR Inactive, message gives details.

8.2.17 Test 32: 690x0 SVGA Controller Test

This BIST checks the operation of an Intel (Chips and Technologies) 69000 or 69030 Graphics Controller. Only the default console device is tested; for slave boards, the controller that would have been elected the default will be tested.

First a register access test is performed to the hardware cursor high and low registers; this uses ports 03D4h and 03D5h. Then the linear video memory is auto-sized and the result reported. Finally, the linear video memory is tested via 16-bit accesses using the address offset as a pattern.

Error codes:

0401h - no 690x0 VGA console available,
0402h - failed register test with 055h pattern,
0403h - failed register test with 0AAh pattern,
0405h - failed linear memory pattern test.

NOTE This test will operate correctly even if a serial port is used as the VSA console. A VGA monitor is not required for this test.

8.2.18 Test 33: Universe PCI VME Test

This BIST performs a basic functional check on the Universe II. First, a write-then-read test is performed on PCI3 slave BS register. Then interrupt generation is tested by performing a posted write to non-existent VME address.

Error codes:

0402h - error in read/write test,
0406h - no interrupt or spurious interrupt during test.

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

8.2.19 Test 34: Universe PCI Config Utility

This pseudo test configures a universe PCI slave image register for off-board VME accesses.

The following parameters are required:

Slave to program (Default = 3),

Lower Address (Default = 90000000h),

Upper Address (Default = 91000000h),

Translation Offset (Default = 0),

Control Register Value (Default = 80820000h).

This test does not fail.

8.2.20 Test 35: Universe VME Config Utility

This pseudo test configures a universe VME slave image register so the board responds to VME accesses.

The following parameters are required:

Slave to program (Default = 3),

Lower Address (Default = 90000000h),

Upper Address (Default = 91000000h),

Translation Offset (Default = 0),

Control Register Value (Default = 80520000h).

This test does not fail.

8.2.21 Test 36: VME Bus Byte Swapping

This BIST is used to test the Hardware Byte Swapping Features when reading / writing the VME bus. The test requires another board in the VME rack to test with.

The test works by configuring a Universe PCI slave image for VME bus access, writing known Byte, Word and Double word values to the slave VME board then enabling Byte Swapping in the control register.

The data is read back and compared with expected values, any discrepancies are reported along with the error codes.

The Universe slave image and Byte swapping is disabled at the end of the test.

When testing is performed using a Concurrent Technologies soak master, this test will operate as a co-operating BIST where two boards perform the Byte Swapping test on each others memory simultaneously.

Error Codes:

0400h - Error comparing byte 0
0401h - Error comparing byte 1
0402h - Error comparing byte 2
0403h - Error comparing byte 3
0410h - Error comparing word 0
0411h - Error comparing word 1
0420h - Error comparing double word

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

8.2.22 Test 38: SCSI Processor Self-Test

This BIST uses the self-test features of the SCSI processor to check its operation and the interface with the baseboard.

The controller is reset and configured for the local bus interface. Register access is verified by writing patterns to the SCRATCH register and verifying them. Then, the DMA and SCSI FIFOs are tested using the controller's internal test registers.

Finally, the processor is made to execute a simple script to perform a memory- to-memory transfer and interrupt the CPU. The content of the destination buffer is compared with the source data and the correct operation of the interrupt signal is verified.

Error codes:

0300h - An error has occurred. Accompanying message gives further details.

8.2.23 Test 39: SCSI Interface

This cooperating BIST requires the use of two SCSI controller boards and can only be run in conjunction with a CCT soak test master. The two controllers must be the only connections to the SCSI bus.
During the test, one board assumes the role of master and writes a sliding one's pattern, first to the SCSI control bus, then to the SCSI data bus. For each test the slave board reads and verifies the sequence.

Error Codes:

0401h - CCT test master not detected by agent
0402h - Master board did not detect slave
0403h - Master SCSI control bus test failed
0404h - Master SCSI data bus test failed
0405h - Slave board did not detect master
0406h - Slave SCSI control bus test failed
0407h - Slave SCSI data bus test failed

8.2.24 Test 48: SCSI Hard Disk Drive Ready

A drive ready test will be performed on the hard disk drive attached to the SCSI bus. A particular make and type of drive is assumed by the test, as this is intended to be a simple bench test for manufacturing purposes.

The drive's logical unit number is 0 and its SCSI ID will default to 2; however, this can be overridden when invoking the test by supplying the required SCSI ID as the single parameter.

Error codes:

0300h - Test failed: see message for details.

8.2.25 Test 49: SCSI Floppy Disk Drive Ready

A drive ready test will be performed on the floppy disk drive attached to the SCSI bus. A particular make and type of drive is assumed by the test, as this is intended to be a simple bench test for manufacturing purposes.

The drive's logical unit number is 0 and its SCSI ID will default to 0; however, this can be overridden when invoking the test by supplying the required SCSI ID as the single parameter.

Error codes:

0300h - Test failed: see message for details.

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

8.2.26 Test 50: SCSI Tape Drive Ready

A drive ready test will be performed on the tape drive attached to the SCSI bus. A particular make and type of drive is assumed by the test, as this is intended to be a simple bench test for manufacturing purposes.

The drive's logical unit number is 0 and its SCSI ID will default to 6; however, this can be overridden when invoking the test by supplying the required SCSI ID as the single parameter.

Error codes:

0300h - Test failed: see message for details.

8.2.27 Test 51: SCSI Hard Disk Drive Read Test

A device ready test will be performed on the hard disk drive attached to the SCSI bus, followed by a read test.

The drive's logical unit number is 0 and its SCSI ID will default to 2; however, this can be overridden when invoking the test.

The following parameters may be supplied to the test:

1. Printout: flag to indicate output preference:

   0 - no output (default)

   1 - output to screen

   2 - output to screen in paged mode

2. Target ID: the SCSI ID of the drive under test, (default = 2).
3. Start Block: the block number from which to start reading, (default = 0).
4. Block count: the number of blocks to read, 1-255, (default = 1).

Error codes:

0300h - Test failed: message gives details of error.

8.2.28 Test 52: SCSI Floppy Disk Read Test

A device ready test will be performed on the floppy disk drive attached to the SCSI bus, followed by a read test.

The drive's logical unit number is 0 and its SCSI ID will default to 0; however, this can be overridden when invoking the test.

The following parameters may be supplied to the test:

1. Printout: flag to indicate output preference:

   0 - no output (default)

   1 - output to screen

   2 - output to screen in paged mode

2. Target ID: the SCSI ID of the drive under test, (default = 2).
3. Start Block: the block number from which to start reading, (default = 0).
4. Block count: the number of blocks to read, 1-255, (default = 1).

Error codes:

0300h - Test failed: message gives details of error.

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

8.2.29 Test 53: SCSI Tape Drive Read Test

A device ready test will be performed on the tape drive attached to the SCSI bus, followed by a read test.

The drive's logical unit number is 0 and its SCSI ID will default to 6; however, this can be overridden when invoking the test.

The following parameters may be supplied to the test:

1. Printout: flag to indicate output preference:

   0 - no output (default)
   1 - output to screen
   2 - output to screen in paged mode

2. Target ID: the SCSI ID of the drive under test, (default = 2).
3. Start Block: the block number from which to start reading, (default = 0).
4. Block count: the number of blocks to read, 1-255, (default = 1).

Error codes:

0300h- Test failed: message gives details of error.

8.2.30 Test 54: SCSI Inquiry

This BIST issues a SCSI Inquiry command to each possible device on the SCSI bus. For each device found, it prints the target ID and logical unit number of the device, together with the device type, the state of the RMB bit and the ANSI version number.

If additional information is supplied by the device, the BIST also prints the vendor, product and revision strings for the device.

Error codes:

0300h - SCSI Inquiry command failed: details are given in message.

8.2.31 Test 55: SCSI Reset Test

This BIST asserts the SCSI bus reset signal and verifies that this has happened.

This test does not run at power-up, but may be invoked thereafter by a Master Test Handler.

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

8.2.32 Test 56: IDE Controller Test

This BIST checks the operation of the embedded IDE controller that forms part of the PIIX4 south bridge.

This test consists of a number of sub-tests, which can be selected via a command line parameter. If the BIST is invoked without parameters, only those tests that exercise the controller are performed. The following sub-tests are available, (D) indicates a test run by default:

0 - Run default tests (D)

1 - Register access test (D)

2 - Controller diagnostics test (D)

3 - Identify disk drive

8.2.32.1 Register Access Test

This sub-test performs a write-then-read check on the controllers internal registers. The sectors-per-track, sector-number and low-cylinder-count registers are tested.

8.2.32.2 Controller Diagnostics Test

This sub-test invokes the IDE controller's internal diagnostic check. If the check fails, the diagnostic error code is displayed.

8.2.32.3 Identify Disk Drive

This sub-test uses the 'Identify Drive' command to interrogate the controller on the disk drive. The manufacturers model name, the physical geometry and the highest supported PIO, DMA and UDMA modes are displayed.

Error codes:

0400h - register test miscompare,
0401h - controller diagnostics error,
0402h - drive identify generated an error,
04FFh - disk controller not found.

NOTE An IDE disk drive must be connected for the BIST to operate.

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

8.2.33 Test 57: SCSI Fixture Test

This BIST checks the operation of the on-board SCSI controller by means of an external test fixture. This fixture is identified as "TF0169".

There are no sub-commands or parameters relevant to this test.

The fixture tests the following features of the SCSI interface:

Register address lines
Chip select lines
Data lines
DMA channels
Interrupt lines

Error codes:

04C0h - SEQUENCE: Parity Error
04C1h - SBCL: Read error
04C2h - SBDL: Read error
04C3h - TERM.POWER: Parity Error
04C4h - SBCL: Read Error – expected 55
04C5h - SBDL: Read Error – expected 11 or 22
04C6h - Terminal Power is Absent
04C7h - FINAL: Parity Error
04C8h - SBCL: Read Error – expected 00
04C9h - SBDL: Read Error – expected 00
04FFh - Test Fixture Not Found

8.2.34 Test 58: IDE Fixture Test

This BIST checks the operation of the on-board IDE controller by means of an external test fixture. This fixture is identified as "TF0169".

There are no sub-commands or parameters relevant to this test.

This fixture tests the following features of the IDE interface:

Register Address Lines
Chip Select Lines
Data Lines
DMA Channels

Error codes:

The error codes for this BIST are returned as a range of error codes, the least significant digit representing the point at which it failed in the test sequence. For example 0436h identifies that DMA failed during the transfer of byte 6.

040yh - Register Test Failed on Byte y
041yh - Chip Select Failed on Byte y
042yh - Data Test Failed on Byte y
043yh - DMA Failed on Byte y
04FFh - Test Fixture Not Found.

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

8.2.35 Test 64: PC Keyboard Test

This BIST performs checks on the keyboard controller, the test also determines whether a keyboard is present.

First, the keyboard controller's output buffer is flushed and a 'keyboard present' test is performed. The keyboard controller is then enabled and initialized and if successful, the keyboard controller's self test and interface test are performed. Finally a keyboard interrupt is generated and verified.

Error codes:

0401h - Time-out trying to flush keyboard controller buffer.
0402h - Keyboard controller did not read keyboard enable command.
0403h - Keyboard controller did not read mode command.
0404h - Keyboard controller did not read mode command data.
0405h - Self test command not read.
0406h - Time-out waiting for self test result.
0407h - Self test fail.
0408h - Keyboard controller not ready after self test.
0409h - Interface test command not read.
040Ah - Time-out waiting for interface test result.
040Bh - Interface test fail. Keyboard clock line stuck high.
040Ch - Interface test fail. Keyboard clock line stuck low.
040Dh - Interface test fail. Keyboard data line stuck high.
040Eh - Interface test fail. Keyboard data line stuck low.
0410h - Error while trying to cause an interrupt.
0411h - No keyboard interrupt.
0412h - Wrong interrupt received.

8.2.36 Test 67: Printer Port Test

This BIST checks the PC compatible printer port using a Concurrent Technologies printer test fixture.

A marching 1/0 test is performed on the PORTC and PORTA registers. Then a marching 1/0 test is performed on PORTA via test fixture. Finally, a 1/0 test is performed on PORTB driven by PORTA and PORTC, via the test fixture.

Error codes:

0450h = PORTC marching '0' failed
0451h = PORTC marching '1' failed
0452h = PORTA marching '0' failed
0453h = PORTA marching '1' failed
0454h = PORTA marching '0' via test fixture failed
0455h = PORTA marching '1' via test fixture failed
0458h = PORTB write 0h failed
0459h = PORTB write STB = 0 failed
045Ah = PORTB write PINIT = 1 failed
045Bh = PORTB write PSEL = 1 failed
045Ch = PORTB write PORTA D0 = 1 failed
045Dh = PORTB write PORTA D0 = 1 failed

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

8.2.37 Test 68: Real Time Clock Test

This BIST tests the PC compatible, real time clock. The BIST provides a number of sub-tests, which are selected by a command parameter. If no parameter is supplied the current time and date is displayed, the interrupt signal is tested and the non-destructive NVRAM test performed.

The sub-tests options are:

0 - Set date and time. Followed by:

Hour (0 - 23),

Minute (0 - 59),

Day (1 - 31),

Month (1 - 12),

Year (0 - 99).

1 - Display time and date, then do interrupt test,

2 - Clear contents of NVRAM,

3 - Display contents of NVRAM,

4 - Non-destructive read/write test of NVRAM.

The RTC periodic interrupt is allowed to interrupt twice to test that the interrupt is acknowledged correctly.

The read/write test checks each location of NVRAM (excluding the RTC registers). Each address is tested first with 55h, then with AAh. The contents of NVRAM is saved and restored around the test.

Error codes:

0300h = Fail. Message will describe failure in some detail.

04xyh

x 0 - No interrupt occurred,

1 - Wrong interrupt occurred.

y - Test numbers:

2 = Timer 0 interrupt

4 = Timer 2 interrupt

5 = First RTC periodic interrupt

6 = Second RTC periodic interrupt

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

8.2.38 Test 69: Digital 21143 100MHz LAN

This BIST exercises a 21143 Ethernet Controller on any PCI device or bus number. The sub-test number and device to test are specified as BIST parameters.

The following sub-tests are available:

1. All default tests (1, 2 and 5),
2. Device checks,
2. Internal loopback,
3. External loopback 10MB/s,
4. External loopback 100MB/s,
5. Interrupt test,
6. Test SROM,
7. Display CSR registers,
8. Display Ethernet address,
9. Display SROM contents.

If no parameters are entered then default tests (1,2 and 5) are performed on all 21143 devices located through a PCI bus scan.

Default operating parameters are automatically loaded from the on-board serial EEPROM.

The IEEE/ANSI 802.3 station address is extracted from the serial EEPROM. The configured station address is reported via the console.

Internal and external loopback testing is performed by transferring a frame and verifying the received pattern.

The device's timer is used to generate an interrupt and baseboard recognition of the correct signal is verified.

Error codes:

0400h - Could not find the specified 21143,
0401h - The specified sub test number is not valid,
0402h - No PCI memory address has been assigned,
0403h - No PCI interrupt line has been assigned,
0404h - The device stop command failed,
0405h - PCI bus error occurred,
0410h - Cannot set CSR3 to 0,
0411h - Cannot set CSR3 to 1,
0412h - Cannot set CSR4 to 0,
0413h - Cannot set CSR4 to 1,
0420h - Received data is the wrong length,
0421h - Received data does not match transmitted data,
0422h - Transmitter time-out,
0423h - Receiver time-out,
0424h - Transmit descriptor error,
0425h - Receive descriptor error.
0430h - No interrupt has been detected,
0431h - An interrupt has occurred on the wrong channel,
04F0h - An error occurred while erasing the serial EEPROM,
04F1h - An error occurred while writing to the serial EEPROM,
04F2h - Serial EEPROM CRC error.

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

8.2.39 Test 70: Maxim 1617 Thermal Sensor Test

This BIST checks the operation of the Maxim 1617 Thermal Sensor.

This test consists of a number of sub-tests, which can be selected via a command line parameter. If the BIST is invoked without parameters, only basic diagnostics and CPU over-heat are checked. The CPU over-heat temperature is preset to 95°C. The following sub-tests are available; (D) indicates a test run by default; (B) indicates a test which performs a *basic functionality* test first.

0 – Temperature readout (B) (D)

1 – Set alarms (B)

2 – Change update frequency (B)

3 – Full readout

8.2.39.1 Basic Functionality

The basic functionality tests perform the following checks on the Maxim 1617 Thermal Sensor:

The CPU sensor is not *open-circuit.*

The CPU sensor is not *short-circuit.* Should this error occur, the CPU sensor will always read 0°C.

The CPU sensor is not connected to Vcc. Should this error occur both the ambient and CPU sensors will always read 127°C.

8.2.39.2 Temperature Readout

This sub-test will perform a basic functionality test, display the current temperature readings and finally check that the CPU has not breached the 95°C alarm. Should this alarm be triggered or basic diagnostics fail, the BIST will report an error.

8.2.39.3 Set Alarms

This option allows the user to program the four alarms: Ambient Low; Ambient High, CPU Low and CPU high. The range on each alarm is –65°C to 127°C. These temperatures may be specified as BIST parameters in the order above.

As BISTs do not provide support for the minus sign, negative numbers should be entered by replacing the "-" symbol with a "9" (I.e. "-34" becomes "934"). 0°C is valid but, for the CPU sensor, may be confused with a short-circuit. 127°C is valid but, on both sensors, may be confused with a short to Vcc.

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

8.2.39.4 Change Update Frequency

This option allows the user to change the update frequency of the Maxim 1617 Thermal Sensor. All possible options are listed below:

Value	Update Frequency (Hz)
0	0.0625
1	0.125
2	0.25
3	0.5
4	1
5	2
6	4
7	8

Alarms will only trigger when an update occurs. Should there be a temperature spike between readings it will not trigger an alarm. If this is a problem raise the update frequency to 8Hz.

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

8.2.39.5 Full Readout

This option reads and displays the data currently available from the Maxim 1617 Thermal Sensor. The display is in the following format.

```
!ALERT mask   : 1          S/w standby   : 0
Conv. rate    : 0.0625Hz   Chip busy     : 0
CPU OPEN      : 0          CPU SHORT     : ?       CPU VCC     : ?
Temp. Amb.    : xxx^C      Temp. CPU     : xxx^C
Triggered alarms:
*** Ambient High Threshold ***
*** Ambient Low Threshold ***
*** CPU High Threshold ***
*** CPU Low Threshold ***
```

!ALERT mask	Hardware alarm mask. 1=enabled, 0=disabled
S/w standby	Software standby mode, 1=enabled, 0=disabled
Conv. Rate	Frequency (Hz) at which the temperature readings are updated.
Chip busy	The Maxim 1617 is currently updating.
CPU OPEN	The CPUs thermal sensor is currently showing as an open circuit.
CPU SHORT	The CPUs thermal sensor is currently reading 0°C.
CPU VCC	Both the CPU and Maxim thermal sensors are currently reading 127°C.
Temp. Amb.	Reading in degrees centigrade (°C) of the Maxim 1617 Thermal Sensor chip.
Temp. CPU	Reading in degrees centigrade (°C) of the CPU.
Triggered alarms	This is a list of which alarms have activated since the last check.

There are four alarms which may be triggered. Every pair of readings taken by the Maxim 1617 Thermal Sensor are compared against the thresholds and will flag any relevant alarms.

Error codes:

040Ah - Read or write request to an invalid device
040Bh - Bad data type requested.
040Ch - Multiple termination conditions exist.
040Dh - Invalid command line parameter.
0411h - Process was KILLed by another program.
0412h - Bus collision.
0414h - Transaction error. This is caused by either:

Illegal Command
Unclaimed cycle (Host initiated)
Host device time-out

0418h - The device failed to respond in the preset time. The preset time-out is 10 seconds per command.
0421h - The CPUs thermal sensor is currently showing as an open circuit.
0422h - The CPUs thermal sensor is currently reading 0°C. A reading of 0°C is unlikely for this motherboard, therefore it is assumed to be the error condition which causes this reading.
0423h - Both the CPU and Maxim thermal sensors are currently reading 127°C. This reading is unlikely for this motherboard, therefore it is assumed to be the error condition which causes this reading.
04FFh - The CPU has breached the 95°C safety check.

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

8.2.40 Test 71: DEC 21143 Interface Test

This BIST verifies the operation of the external Ethernet interface of a 21143 Ethernet controller. The test will only run with CCT's Soak Test Master and in conjunction with a second, Ethernet equipped board.

The test exchanges data packets between two boards. One board is designated 'master' by the test controller and will transmit first. The received data is checked against expected results.

Error codes:

0400h - Could not find the specified 21143,
0402h - No PCI memory address has been assigned,
0403h - No PCI interrupt line has been assigned,
0404h - The device stop command failed,
0440h - No test Master available
0441h - Timeout during test
0442h - Data mismatch (board is Master)
0443h - Data mismatch (board is Slave)

8.2.41 Test 80: SCSI Based PMC Site Test

This BIST uses a CCT SC PMC/825 or SC PMC/875 PMC SCSI module to verify the operation of any PMC sites connected to the board. Both baseboard and carrier-based PMC sites are tested.

Error codes:

0401h - could not find a SC PMC/825 or SC PMC/875 PMC SCSI module to conduct the test.
0402h - a module failed the test. The associated error message identifies the module and reason.

8.2.42 Test 85: Floppy Disk Drive Test

This BIST checks the operation of the PC compatible floppy disk controller and associated hardware. The BIST comprises a number of sub-tests, of which only the Floppy Controller test is run by default. The BIST operates on drive A: by default, however drive B: can be specified by a BIST parameter.

8.2.42.1 Controller Access Test

This sub-test checks access to the floppy disk controller hardware. No disk drive is required for this test.

8.2.42.2 Diskette Access Test

This sub-test checks access to a floppy disk drive by reading a single sector from a floppy disk. The content of the disk is not important; the first 256 bytes of the sector will be displayed on screen.

8.2.42.3 Disk Checksum Test

This sub-test reads the entire contents of a floppy disk and computes the byte checksum (32-bit sum-of-bytes). The computed checksum value is displayed allowing different test disks to be used.

Error codes:

0420h - Failure during Floppy Controller test.
0440h - Failure during Disk Access test.
0480h - Failure during Disk Verify test.
04FFh - Invalid BIST parameter supplied.

These values are modified using the following sub-codes to identify the cause of the error.

01h - Error during controller command phase
02h - Error writing data to the controller
04h - Error reading data from the controller
05h - Wrong interrupt received
06h - No interrupt received

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

8.2.43 Test 101: Display Memory Utility

This BIST allows any area of the target board's local memory to be examined and displayed by the test master. This utility requires command-line parameters to function correctly, so it should only be run in an interactive manner by a local or remote test master.

The parameters are:

start address of memory area (default 0),

length of memory area in bytes (default 10h),

data type (1 for byte, 2 for word, and 4 for dword)(default 1).

The results are displayed as hexadecimal values.

This is not a true BIST, but merely provides a utility function, and so always returns a PASS status.

8.2.44 Test 102: Fill Memory Utility

This BIST allows any area of the target board's local RAM to be filled with a constant value by the test master. This utility requires command-line parameters to function correctly, so it should only be run in an interactive manner by a local or remote test master.

The parameters are:

start address of memory area (default 0),

data type (1 for byte, 2 for word, and 4 for dword) (default 1),

length of RAM area in bytes (default 1),

constant value with which to fill the area (default 0).

This is not a true BIST, but merely provides a utility function, and so always returns a PASS status.

8.2.45 Test 103: I/O Read Utility

This BIST allows examination of any I/O register on the target board. This utility requires command-line parameters to function correctly, so it should only be run in an interactive manner by a local or remote test master.

The parameters are:

16-bit I/O address (default 0),

data type (1 for byte, 2 for word, and 4 for dword) (default 1),

increment value for the port address (default 1),

number of times to perform an I/O read (default 1),

The result is displayed as a hexadecimal value.

This is not a true BIST, but merely provides a utility function, and so always returns a PASS status.

8.2.46 Test 104: I/O Write Utility

This BIST allows modification of any I/O register on the target board. This utility requires command-line parameters to function correctly, so it should only be run in an interactive manner by a local or remote test master.

The parameters are:

16-bit I/O address (default 0),

value to write to register (default 0),

data type (1 for byte, 2 for word, and 4 for dword) (default 1),

increment value for the port address (default 1),

number of times to perform an I/O write (default 1).

This is not a true BIST, but merely provides a utility function, and so always returns a PASS status.

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

8.2.47 Test 105: Interconnect Read Utility

This BIST allows a local interconnect read to be performed on the target board. This utility requires command-line parameters to function correctly, so it should only be run in an interactive manner by a local or remote test master.

The parameter is:

interconnect register number (16-bit value).

The result is displayed as a hexadecimal value.

This is not a true BIST, but merely provides a utility function, and so always returns a PASS status.

8.2.48 Test 106: Interconnect Write Utility

This BIST allows a local interconnect write to be performed on the target board. This utility requires command-line parameters to function correctly, so it should only be run in an interactive manner by a local or remote test master.

Because this operation is carried out as a local access on the target board, it allows a remote agent to write to interconnect registers for which it would normally have read-only access.

The parameters are:

interconnect register number (16-bit value),

new register value (8-bit value).

This is not a true BIST, but merely provides a utility function, and so always returns a PASS status.

8.2.49 Test 107: Cache Control Utility

This BIST allows the status of DRAM and EPROM caching on the target board to be interrogated or configured. If the utility is invoked without parameters, the default action is to display the state of DRAM and EPROM caching.

The available options are:

1. disable DRAM caching,
2. enable DRAM and EPROM caching,
2. toggle DRAM caching state,
3. report DRAM and EPROM caching state (default),
4. disable EPROM caching.

NOTE In normal operation, EPROM Caching should not be disabled. When EPROM caching is disabled, ROM-based timing loops are disrupted, which can cause BISTs to time-out or fail.

8.2.50 Test 120: PCI Configuration Utility

This BIST will display, for each device on the PCI bus, the vendor identification number, device identification number and the device revision number. For example:

Bus	Dev	Func	Vendor ID	Device ID	Revision
00	00	00	8086	7192	00
00	06	00	1000	000C	01
00	07	00	8086	7111	02
00	07	01	8086	7112	01
..	..	..	....	....	..

NOTE The revision numbers may change.

This is not a true BIST, but merely provides a utility function, and so always returns a PASS status.

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

8.2.51 Test 121: PCI Read Utility.

This BIST allows PCI configuration registers to be examined on the target board. This utility requires command-line parameters to function correctly, so it should only be run in an interactive manner by a local or remote test master.

The parameters are:

Device number	= 0 to 31 - default = 0
Register Offset	= 0 to 255 - default = 0
Data Type	= 1 (Byte), 2 (Word) or 4 (Dword) - default = 2
Length	= number of bytes, words, Dwords
Bus Number	= 0 to 255 - default = 0
Function Number	= 0 to 7 - default = 0.

The result is displayed as a hexadecimal value.

This is not a true BIST, but merely provides a utility function, and so always returns a PASS status.

8.2.52 Test 122: PCI Write Utility.

This BIST allows PCI configuration registers to be modified on the target board. This utility requires command-line parameters to function correctly, so it should only be run in an interactive manner by a local or remote test master.

The parameters are:

Device number	= 0 to 31 - default = 0
Register Offset	= 0 to 255 - default = 0
Value to write	= ? (Default 0)
Data Type	= 1 (Byte), 2 (Word) or 4 (Dword) - default = 2
Length	= number of bytes, words, Dwords
Bus Number	= 0 to 255 - default = 0
Function Number	= 0 to 7 - default = 0.
Verify	= 0 (no verify), 1 (verify by reading back) - default = 0

8.2.53 Test 126: Display Board Configuration

This pseudo-test displays the board configuration as seen by the processor. It performs no actual testing, and assumes a basic level of operation by the board. The test does not fail.

The test does not run at power-up, though it can be invoked thereafter by a Master Test Handler.

8.2.54 Test 127: Retrieve BIST Information

This BIST is intended to provide information only for factory testing of the board. It does not fail, but returns information in an encoded form for use during automatic testing prior to shipment.

The test is normally run only by a Test Master.

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

Appendix A

A.1 Functional Specification

Processor:	• 433 or 566MHz Celeron, or 700 or 850MHz Pentium III with 32 Kbyte Level 1 cache, defined by order code.
Level 2 Cache:	• 128 Kbytes (Celeron) or 256 Kbytes (Pentium III) on-die RAM operating at core frequency.
Memory:	• PC BIOS Flash EPROM 32-pin PLCC JEDEC socket which accepts 29F040 to give 512 Kbytes. • SDRAM up to 384 Mbytes defined by order number. Processor burst and cache support.
Interfaces:	• VME64 interface utilizing the 91C142 UNIVERSE II PCI to VME bridge device. • One RS232 serial channel using a 16550 compatible UART. Baud rate generator source clock of 24MHz. • Wide or Narrow Ultra SCSI interface implemented with 53C875, via P2. Maximum Wide SCSI bus transfer rate of 14 Mbytes/s asynchronous and 40 Mbytes/s synchronous. Active SCSI bus termination and signal negation. • EIDE/Ultra DMA-33 interface via P2. Maximum transfer rate of 33 Mbytes/s. • Floppy disk interface supporting up to 1 Mbyte/s transfer rates and 2 floppy disk drives. • Single parallel printer interface, via P2. • USB interface via front panel connector and P2. Both 1.5 and 12 Mbit/s interfaces supported. • A single width PMC site supporting a 32 bit PCI interface with 5V signaling. Both 5V and 3.3V power rails are supported. • A PCI expansion interface supporting a 32 bit PCI interface with 5V signaling. Both 5V and 3.3V power rails are supported. Allows the addition of two single PMC cards. Requires an adaptor module (not supplied). • PS/2 compatible keyboard interface. • PS/2 compatible mouse interface. • Ethernet interface using 21143 PCI to Ethernet device. 10 Mbit/s or 100 Mbit/s RJ45 connection via front panel. • External Reset input via P2.
Peripherals:	• an Intel PIIX4E device provides two 82C59 compatible PICs, 82C54 compatible PIT, DMA controller, BIOS timer and PC-AT Real Time Clock. • No DiskOnChip component fitted.

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

A.2 Connectors

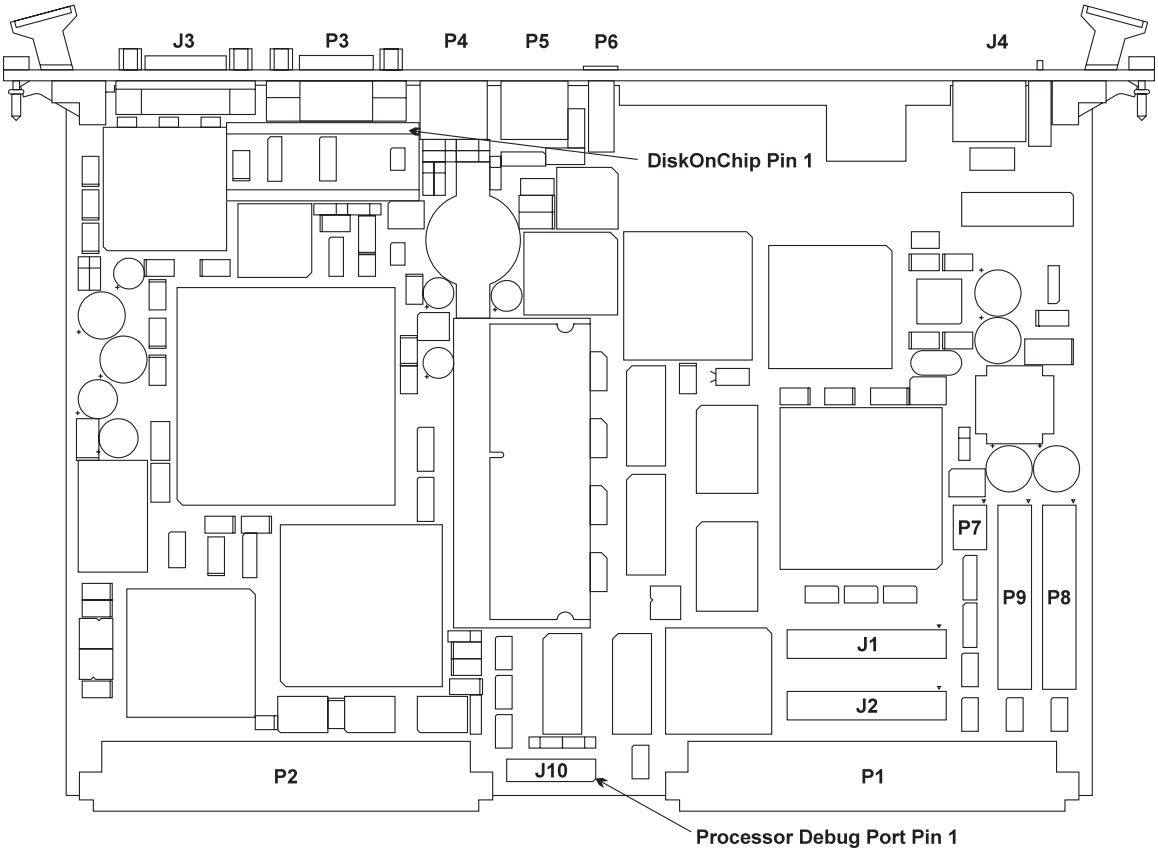


Figure A-1 Connector Layout

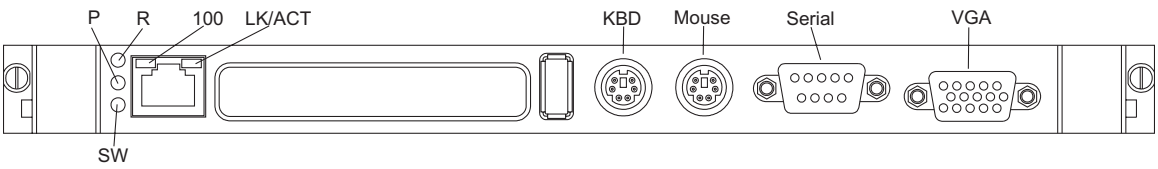


Figure A-2 Front Panel Connector Layout

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

A.2.1 Serial Interface Pin-Outs

The pin-out of the RS232 connector is as follows:

Pin No.	Signal Name	Direction
1	CD0 - Carrier Detect	Input to board
2	RX0 - Rx Data	Input to board
3	TX0 - Tx Data	Output from board
4	DTR0 - Data Terminal Ready	Output from board
5	GND	-
6	DSR - Data Set Ready	Input to board
7	RTS0 - Request To Send	Output from board
8	CTS0 - Clear To Send	Input to board
9	RI0 - Ring Indicator	Input to board

Table A-1 Serial Interface Pin-outs - Channel 0 (COM1)

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

A.2.2 VME Interface (P1) Pin-outs

The VME interface connector P1 consists of a 160-pin connector with pins assigned as follows:

Pin No.	Row Z	Row A	Row B	Row C	Row D
1	-	D00	$\overline{\text{BBSY}}$	D08	-
2	Ground	D01	$\overline{\text{BCLR}}$	D09	Ground
3	-	D02	$\overline{\text{ACFAIL}}$	D10	-
4	Ground	D03	$\overline{\text{BG0IN}}$	D11	-
5	-	D04	$\overline{\text{BG0OUT}}$	D12	-
6	Ground	D05	$\overline{\text{BG1IN}}$	D13	-
7	-	D06	$\overline{\text{BG1OUT}}$	D14	-
8	Ground	D07	$\overline{\text{BG2IN}}$	D15	-
9	-	Ground	$\overline{\text{BG2OUT}}$	Ground	-
10	Ground	SYSCLK	$\overline{\text{BG3IN}}$	$\overline{\text{SYSFAIL}}$	$\overline{\text{GAP}}$
11	-	Ground	$\overline{\text{BG3OUT}}$	$\overline{\text{BERR}}$	$\overline{\text{GA0}}$
12	Ground	$\overline{\text{DS1}}$	$\overline{\text{BR0}}$	$\overline{\text{SYSRESET}}$	$\overline{\text{GA1}}$
13	-	$\overline{\text{DS0}}$	$\overline{\text{BR1}}$	$\overline{\text{LWORD}}$	-
14	Ground	$\overline{\text{WRITE}}$	$\overline{\text{BR2}}$	AM5	$\overline{\text{GA2}}$
15	-	Ground	BR3	A23	-
16	Ground	$\overline{\text{DTACK}}$	AM0	A22	$\overline{\text{GA3}}$
17	-	Ground	AM1	A21	-
18	Ground	AS	AM2	A20	GA4
19	-	Ground	AM3	A19	-
20	Ground	$\overline{\text{IACK}}$	Ground	A18	-
21	-	$\overline{\text{IACKIN}}$	-	A17	-
22	Ground	$\overline{\text{IACKOUT}}$	-	A16	-
23	-	AM4	Ground	A15	-
24	Ground	A07	$\overline{\text{IRQ7}}$	A14	-
25	-	A06	$\overline{\text{IRQ6}}$	A13	-
26	Ground	A05	$\overline{\text{IRQ5}}$	A12	-
27	-	A04	$\overline{\text{IRQ4}}$	A11	-
28	Ground	A03	$\overline{\text{IRQ3}}$	A10	-
29	-	A02	$\overline{\text{IRQ2}}$	A09	-
30	Ground	A01	$\overline{\text{IRQ1}}$	A08	-
31	-	-12V	-	+12V	Ground
32	Ground	+5V	+5V	+5V	-

Table A-2 VME Interface Pin-outs

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

A.2.3 Auxiliary Connector (P2) Pin-outs (Wide SCSI, Panel Link & EIDE)

The auxiliary connection P2 consists of a 160-pin connector. When the board is configured at the factory for a Wide SCSI and Panel Link interface using all the pins on this connector, the pin assignments are as shown in Table A-3.

Pin No.	Row Z	Row A	Row B	Row C	Row D
1	GPOUT	SCSID0	+5V	DRVDEN0	-
2	GND	SCSID1	GND	DRVDEN1	-
3	GPIN	SCSID2	RETRY	EXT RST	IDEDD7
4	GND	SCSID3	A24	INDX	IDEDD9
5	IDERST	SCSID4	A25	FDME0	IDEDD6
6	GND	SCSID5	A26	FDS1	IDEDD10
7	IDEDD8	SCSID6	A27	FDS0	IDEDD5
8	GND	SCSID7	A28	FDME1	IDEDD11
9	-	SCSIDP	A29	DIR	IDEDD4
10	GND	GND	A30	STEP	IDEDD12
11	-	GND	A31	WRDATA	IDEDD3
12	GND	GND	GND	WE	IDEDD13
13	-	TERMPWR	+5V	TRK0	IDEDD2
14	GND	GND	D16	WP	IDEDD14
15	-	GND	D17	RDDATA	IDEDD1
16	GND	SCSIATN	D18	HDSEL	IDEDD15
17	-	GND	D19	DSKCHG	IDEDD0
18	GND	SCSIBSY	D20	Reserved	IDEDRQ
19	-	SCSIACK	D21	Reserved	IDEIOW
20	GND	SCSIRST	D22	SCSID8	IDEIOR
21	GDCCK	SCSIMSG	D23	SCSID9	IDEDRDY
22	GND	SCSISEL	GND	SCSID10	IDEDACK
23	GDCDAT	SCSICD	D24	SCSID11	IDEINT
24	GND	SCSIREQ	D25	SCSID12	IDEA1
25	HOT PLUG	SCSIIO	D26	SCSID13	IDEA2
26	GND	Reserved	D27	SCSID14	IDEA0
27	USBD1	Reserved	D28	SCSID15	IDECS1
28	GND	Reserved	D29	SCSIDP1	IDECS0
29	USBD1	TX2	D30	TX0	IDEACT
30	GND	TX2	D31	TX0	GND
31	-	TXC	GND	TX1	GND
32	GND	TXC	+5V	TX1	-

Table A-3 P2 Connector Pin-outs (Wide SCSI, Panel Link and EIDE)

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

A.2.4 Auxiliary Connector (P2) Pin-outs (Narrow SCSI, EIDE & Printer)

The auxiliary connection P2 consists of a 160-pin connector. When the board is configured at the factory for a Narrow SCSI and Printer interface using all the pins of this connector, the pin assignments are as shown in Table A-4.

Pin No.	Row Z	Row A	Row B	Row C	Row D
1	GPOUT	SCSID0	+5V	DRVDEN0	-
2	GND	SCSID1	GND	DRVDEN1	-
3	GPIN	SCSID2	RETRY	EXT RST	IDEDD7
4	GND	SCSID3	A24	INDX	IDEDD9
5	IDERST	SCSID4	A25	FDME0	IDEDD6
6	GND	SCSID5	A26	FDS1	IDEDD10
7	IDEDD8	SCSID6	A27	FDS0	IDEDD5
8	GND	SCSID7	A28	FDME1	IDEDD11
9	-	SCSIDP	A29	DIR	IDEDD4
10	GND	GND	A30	STEP	IDEDD12
11	-	GND	A31	WRDATA	IDEDD3
12	GND	GND	GND	WE	IDEDD13
13	-	TERMPWR	+5V	TRK0	IDEDD2
14	GND	GND	D16	WP	IDEDD14
15	-	GND	D17	RDDATA	IDEDD1
16	GND	SCSIATN	D18	HDSEL	IDEDD15
17	-	GND	D19	DSKCHG	IDEDD0
18	GND	SCSIBSY	D20	IDEIOW	IDEDRQ
19	-	SCSIACK	D21	IDEIOR	IDEIOW
20	GND	SCSIRST	D22	STROBE	IDEIOR
21	Reserved	SCSIMSG	D23	PRT D0	IDEDRDY
22	GND	SCSISEL	GND	PRT D1	IDEDACK
23	Reserved	SCSICD	D24	PRT D2	IDEINT
24	GND	SCSIREQ	D25	PRT D3	IDEA1
25	Reserved	SCSIIO	D26	PRT D4	IDEA2
26	GND	Reserved	D27	PRT D5	IDEA0
27	USBD1	Reserved	D28	PRT D6	IDECS1
28	GND	Reserved	D29	PRT D7	IDECS0
29	USBD1	AFD	D30	PRT ACK	IDEACT
30	GND	ERR	D31	PRT BUSY	GND
31	-	INIT	GND	PERR	GND
32	GND	SELECT I	+5V	SELECT	-

Table A-4 P2 Connector Pin-outs (Narrow SCSI, Printer and EIDE)

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

A.2.5 Auxiliary Connector (P2) Pin-outs (EIDE Only)

The auxiliary connection P2 consists of a 160-pin connector. When the board is configured at the factory for an EIDE interface using only the middle 3 rows (96 pins) of this connector, the pin assignments are as shown in Table A-5.

Pin No.	Row Z	Row A	Row B	Row C	Row D
1	GPOUT	IDERST	+5V	DRVDEN0	-
2	GND	IDEDD8	GND	DRVDEN1	-
3	GPIN	IDEDD7	RETRY	EXT RST	IDEDD7
4	GND	IDEDD9	A24	INDX	IDEDD9
5	IDERST	IDEDD6	A25	FDME0	IDEDD6
6	GND	IDEDD10	A26	FDS1	IDEDD10
7	IDEDD8	IDEDD5	A27	FDS0	IDEDD5
8	GND	IDEDD11	A28	FDME1	IDEDD11
9	-	IDEDD4	A29	DIR	IDEDD4
10	GND	IDEDD12	A30	STEP	IDEDD12
11	-	IDEDD3	A31	WRDATA	IDEDD3
12	GND	IDEDD13	GND	WE	IDEDD13
13	-	IDEDD2	+5V	TRK0	IDEDD2
14	GND	IDEDD14	D16	WP	IDEDD14
15	-	IDEDD1	D17	RDDATA	IDEDD1
16	GND	IDEDD15	D18	HDSEL	IDEDD15
17	-	IDEDD0	D19	DSKCHG	IDEDD0
18	GND	IDEDRQ	D20	IDEIOW	IDEDRQ
19	-	IDEDRDY	D21	IDEIOR	IDEIOW
20	GND	IDEDACK	D22	STROBE	IDEIOR
21	-	IDEINT	D23	PRT D0	IDEDRDY
22	GND	IDEA1	GND	PRT D1	IDEDACK
23	-	IDEA2	D24	PRT D2	IDEINT
24	GND	IDEA0	D25	PRT D3	IDEA1
25	-	IDECS1	D26	PRT D4	IDEA2
26	GND	IDECS0	D27	PRT D5	IDEA0
27	USBD1	IDEACT	D28	PRT D6	IDECS1
28	GND	GND	D29	PRT D7	IDECS0
29	USBD1	AFD	D30	PRT ACK	IDEACT
30	GND	ERR	D31	PRT BUSY	GND
31	-	INIT	GND	PERR	GND
32	GND	SELECTI	+5V	SELECT	-

Table A-5 P2 Connector Pin-outs (EIDE Only)

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

A.2.6 Keyboard Connector (P5) Pin-outs

The keyboard connector is a standard PS/2 6-way mini DIN socket, pin numbering is shown in Figure A-3 and assignment is given in Table A-6.

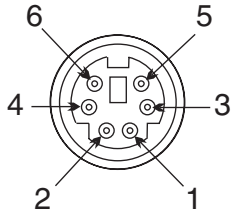


Figure A-3 Keyboard Connector

Pin No.	Signal Name
1	DATA
2	NC
3	GND
4	+5V
5	CLOCK
6	GND
Shell	Chassis GND

Table A-6 Keyboard Connector Pin Assignments

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

A.2.7 Mouse Connector (P4) Pin-outs

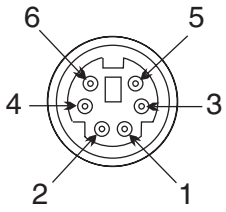


Figure A-4 Mouse Connector

Pin No.	Signal Name
1	DATA
2	NC
3	GND
4	+5V
5	CLOCK
6	GND
Shell	Chassis GND

Table A-7 Mouse Connector Pin Assignments

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

A.2.8 PMC Expansion Carrier Connector

Signal assignments on the PMC Expansion Carrier connectors are shown in Tables A-8 and A-9.

Pin No.	Signal Name	Pin No.	Signal Name
63	-12V	64	-12V
61	GND	62	$\overline{\text{INTA}}$
59	$\overline{\text{INTB}}$	60	$\overline{\text{INTC}}$
57	BUSMODE#A	58	+5V
55	$\overline{\text{INTD}}$	56	PCI-RSVD
53	GND	54	PCI-RSVD
51	$\overline{\text{ACK64}}$†	52	GND
49	GND	50	$\overline{\text{GNT}}$
47	$\overline{\text{REQ}}$	48	+5V
45	V (I/O)	46	AD(31)
43	AD(28)	44	AD(27)
41	AD(25)	42	GND
39	GND	40	$\overline{\text{C/BE(3)}}$
37	AD(22)	38	AD(21)
35	AD(19)	36	+5V
33	V (I/O)	34	AD(17)
31	$\overline{\text{FRAME}}$	32	GND
29	GND	30	$\overline{\text{IRDY}}$
27	$\overline{\text{DEVSEL}}$	28	+5V
25	GND	26	$\overline{\text{LOCK}}$
23	$\overline{\text{SDONE}}$†	24	$\overline{\text{SBO}}$†
21	PAR	22	GND
19	V (I/O)	20	AD(15)
17	AD(12)	18	AD(11)
15	AD(09)	16	+5V
13	GND	14	$\overline{\text{C/BE(0)}}$
11	AD(06)	12	AD(05)
9	AD(04)	10	GND
7	V (I/O)	8	AD(03)
5	AD(02)	6	AD(01)
3	AD(00)	4	+5V
1	GND	2	$\overline{\text{REQ64}}$†

V (I/O) is connected to +5V, † pulled high via 1KOhm resistor

Table A-8 PMC Carrier P9 Connector

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

Pin No.	Signal Name	Pin No.	Signal Name
63	+12V	64	+12V
61	PMCMODE#B	62	GND
59	RESERVED	60	GND
57	GND	58	PCI-RSVD
55	RESERVED	56	PCI-RSVD
53	BUSMODE#2†	54	+3.3V
51	$\overline{\text{RST}}$	52	BUSMODE#3 (GND)
49	3.3V	50	BUSMODE#4 (GND)
47	RESERVED	48	GND
45	AD(30)	46	AD(29)
43	GND	44	AD(26)
41	AD(24)	42	+3.3V
39	RESERVED	40	AD(23)
37	+3.3V	38	AD(20)
35	AD(18)	36	GND
33	AD(16)	34	$\overline{\text{C/BE(2)}}$
31	GND	32	PMC-RSVD
29	$\overline{\text{TRDY}}$	30	+3.3V
27	GND	28	$\overline{\text{STOP}}$
25	$\overline{\text{PERR}}$	26	GND
23	+3.3V	24	$\overline{\text{SERR}}$
21	$\overline{\text{C/BE(1)}}$	22	GND
19	AD(14)	20	AD(13)
17	RESERVED	18	RESERVED
15	AD(08)	16	+3.3V
13	GND	14	RESERVED
11	+3.3V	12	AD(10)
9	+5V	10	RESERVED
7	AD(7)	8	GND
5	GND	6	RESERVED
3	CLK	4	+3.3V
1	GND	2	RESERVED

† pulled high via 1KOhm resistor

Table A-9 PMC Carrier P10 Connector

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

A.2.9 PMC Connector

Signal assignments on the PMC connectors are shown in Tables A-10 and A-11.

Pin No.	Signal Name	Pin No.	Signal Name
1	TCK	2	-12v
3	Ground	4	INTA*
5	INTB*	6	INTC*
7	BUSMODE#1	8	+5V
9	INTD*	10	PCI-RSVD
11	Ground	12	PCI-RSVD
13	CLK	14	Ground
15	Ground	16	GNT*
17	REQ*	18	+5V
19	V (I/O)	20	AD(31)
21	AD(28)	22	AD(27)
23	AD(25)	24	Ground
25	Ground	26	C/BE(3)*
27	AD(22)	28	AD(21)
29	AD(19)	30	+5V
31	V (I/O)	32	AD(17)
33	FRAME*	34	Ground
35	Ground	36	IRDY*
37	DEVSEL*	38	+5V
39	Ground	40	LOCK*
41	SDONE*†	42	SBO*†
43	PAR	44	Ground
45	V (I/O)	46	AD(15)
47	AD(12)	48	AD(11)
49	AD(09)	50	+5V
51	Ground	52	C/BE(0)*
53	AD(06)	54	AD(05)
55	AD(04)	56	Ground
57	V (I/O)	58	AD(03)
59	AD(02)	60	AD(01)
61	AD(00)	62	+5V
63	Ground	64	REQ64*†

V (I/O) is connected to +5V, * denotes active low, † pulled high via 1KOhm resistor

Table A-10 PMC J1 Connector

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

Pin No.	Signal Name	Pin No.	Signal Name
1	+12V	2	TRST*†
3	-	4	-
5	-	6	Ground
7	Ground	8	PCI-RSVD
9	PCI-RSVD	10	PCI-RSVD
11	BUSMODE#2	12	+3.3V
13	RST*	14	BUSMODE#3
15	3.3V	16	BUSMODE#4
17	CCT-RSVD	18	Ground
19	AD(30)	20	AD(29)
21	Ground	22	AD(26)
23	AD(24)	24	+3.3V
25	IDSEL	26	AD(23)
27	+3.3V	28	AD(20)
29	AD(18)	30	Ground
31	AD(16)	32	C/BE(2)*
33	Ground	34	PMC-RSVD
35	TRDY*	36	+3.3V
37	Ground	38	STOP*
39	PERR*	40	Ground
41	+3.3V	42	SERR*
43	C/BE(1)*	44	Ground
45	AD(14)	46	AD(13)
47	Ground	48	AD(10)
49	AD(08)	50	+3.3V
51	AD(07)	52	PMC-RSVD
53	+3.3V	54	PMC-RSVD
55	PMC-RSVD	56	Ground
57	PMC-RSVD	58	PMC-RSVD
59	Ground	60	PMC-RSVD
61	ACK64*†	62	+3.3V
63	Ground	64	PMC-RSVD

* denotes active low, † pulled high via 1KOhm resistor

Table A-11 PMC J2 Connector

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

A.2.10 Ethernet Interface (P11) Pin-outs

The Ethernet Interface is a 8-way RJ45 connector with the following pin-out:

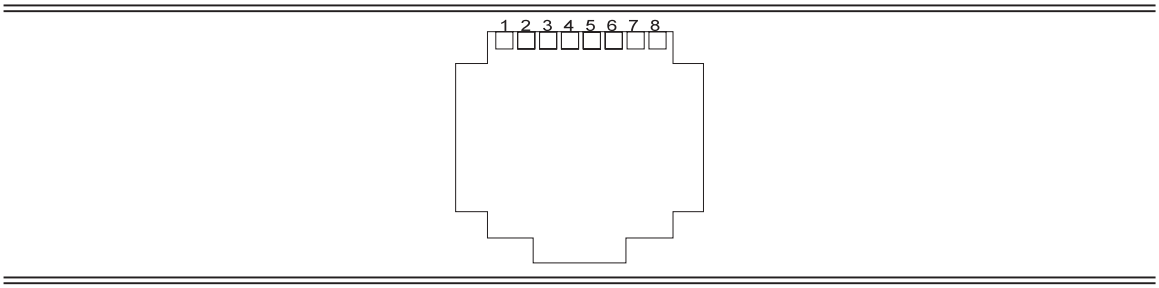


Figure A-5 RJ-45 Connector (Front View)

Pin No	Signal
1	Transmit (+)
2	Transmit (-)
3	Receive (+)
4	Not used
5	Not used
6	Receive (-)
7	Not used
8	Not used

Table A-12 RJ-45 Connector Pin-outs

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

A.2.11 Processor Debug Port (P10) Pin-outs

The processor debug port which is supported by a number of emulator devices is accessible via an Intel specified 30-way header connector with the following pin-out.

Pin No.	Signal
1	CPUReset
2	GND
3	Debug Reset
4	GND
5	CPU TCK
6	GND
7	CPU TMS
8	CPU TDI
9	Pull Up
10	CPU TDO
11	NC
12	CPU TRST
13	GND
14	NC
15	GND
16	CPU REQ
17	GND
18	CPU RDY
19	GND
20	NC
21	GND
22	Pull Up
23	GND
24	NC
25	GND
26	Pull Up
27	GND
28	NC
29	GND
30	Pull Up

Table A-13 30-way Debug Connector Pin-outs

CAUTION Do not use a Pentium II or 433MHz Celeron debug device directly in this connector or damage to the CPU and/or debug device will result. To use a debug device an adapter is required - consult Concurrent Technologies for further details.

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

A.2.12 Universal Serial Bus (P6) Pin-outs

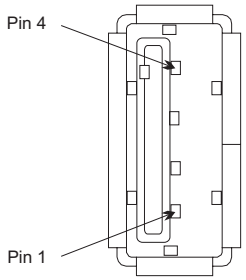


Figure A-6 USB Connector

Pin No.	Signal
1	+5V
2	Data (-)
3	Data (+)
4	GND

Table A-14 USB Pin Assignments

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

A.2.13 VGA (J3) Pin-outs

Pin No.	Signal
1	Analog Red Output
2	Analog Green Output
3	Analog Blue Output
4	N/C
5	Ground
6	Analog Ground
7	Analog Ground
8	Analog Ground
9	N/C
10	Ground
11	N/C
12	DDC Data
13	Horizontal Sync
14	Vertical Sync
15	DDC Clock

Table A-15 VGA Pin Assignments

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

A.3 Environmental Specification

A.3.1 Temperature Range

Operating	0 to 55 degrees Celsius 400LFM air flow, 330LFM for 566MHz Celeron
Storage	-40 to +70 degrees Celsius

A.3.2 Humidity

Operating	10% to 90% non-condensing
Storage	10% to 90% non-condensing

A.4 Dimensions

Height	23.3cm
Depth	16.0cm
Width	2.0cm
Weight	500g

A.5 Electrical Specification

A.5.1 Power Supply Requirements

VOLTAGE (V)	PROCESSOR SPEED	REGULATION	CURRENT (Typical)
+5.0V	433MHz Celeron	+/- 5%	7.0A
+5.0V	566MHz Celeron	+/-5%	5.6A
+5.0V	700MHz Pentium III	+/-5%	7.0A
+5.0V	850MHz Pentium III	+/-5%	7.2A

NOTE This is for a board with 128 Mbytes SDRAM.

NOTE +/- 12V supplies are provided for the PMC interface and are unused if PMC interface expansion is not required.

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

B.1 Memory Addressing Summary

The VP PSE/P3x supports BIOS Flash, SDRAM and OFFBOARD memory. The memory address map has two configurations, real mode and BIOS protected mode.

B.1.1 Real Mode Map

The board will always default to real mode after power on or reset. This mode provides a 1 Mbyte address range with the BIOS Flash memory situated at the CPU reset address.

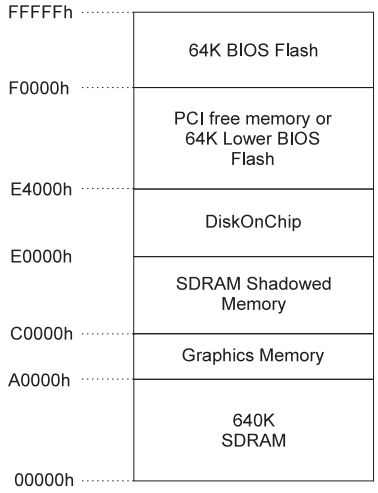


Figure B-1 Real Mode Memory Map

The default real mode map has the 64K lower BIOS Flash disabled. The lower BIOS area appears as PCI free memory.

Addresses above 1 Mbyte will follow the protected mode memory map.

B.1.2 Protected Mode

The BIOS protected mode memory address map gives 4 GBytes address range.

The PC shadow area maybe mapped to SDRAM or configured via the mapping registers of the 443BX and PIIX4E to support standard PC features i.e. video memory or device specific BIOS memory on the PMC modules.

PCI free memory will be used for SCSI, or offboard memory.

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

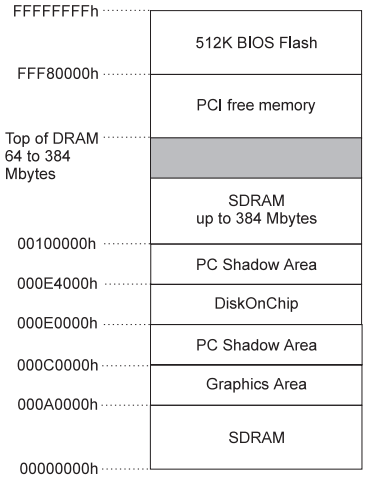


Figure B-2 Protected Mode Memory Map

B.1.3 I/O Addressing Summary

Figure B-3 shows an overall view of the VP PSE/P3x I/O address map. The VP PSE/P3x I/O address map is detailed in Table B-1. This is the normal run time I/O map programmed by the standard factory firmware. Devices which have PCI configuration space are listed in Table B-1.

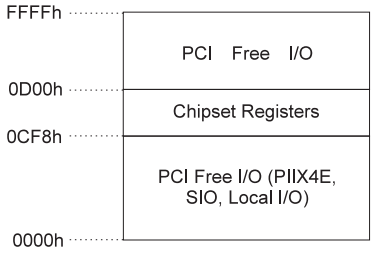


Figure B-3 I/O Space Map

ID	Device
0	North Bridge (443BX)
6	53C875 SCSI Controller
7	PIIX4E
8	69030 Graphics Controller
9	21143 Ethernet Controller
10	Universe II VME Controller
13	PMC Module (if fitted)
19	PMC Expansion (if fitted)

Table B-1 PCI Device IDs

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

B.2 I/O Functions

Location	Description
0000-000Fh (001Fh)	Master DMA Controller (PIIX4E)
0020-0021h (003Fh)	Master Interrupt Controller (PIIX4E)
0040-0043h (005Fh)	Timers 0-2 (82C42)
0060-0064h (006Fh)	Keyboard Controller (PIIX4E)
0070-0077h	RTC/CMOS/NMI-Disable (PIIX4E)
0081-008Fh (008Fh)	DMA Page Registers (PIIX4E)
0092h	Port 92 Register (PIIX4E)
00A0-00A1h (00BFh)	Slave Interrupt Controller (PIIX4E)
00C0-00DFh	Slave DMA Controller (PIIX4E)
0210-0217h	Onboard Control & Status Registers
04D0-04D1h	Interrupt edge/level control (PIIX4E)
0CF8-0CFFh	PCI Configuration Registers
0D00-FFFFh	PCI Free I/O Space

Table B-2 I/O Address Map

Refer to the 443BX and PIIX4E data books for software configurable I/O address mapping.

B.2.1 Interrupt Controllers

Table B-3 lists the interrupt channel assignments.

Controller	Request No.	Priority	Interrupt Source
Master	IRQ0	1	Timer 0
Master	IRQ1	2	Keyboard
Master	IRQ2	3-10	Slave PIC
Slave	IRQ8	3	Real Time Clock / Alarm
Slave	IRQ9	4	Reserved
Slave	IRQ10	5	Reserved
Slave	IRQ11	6	Reserved
Slave	IRQ12	7	Mouse
Slave	IRQ13	8	FPU Error
Slave	IRQ14	9	EIDE
Slave	IRQ15	10	Reserved
Master	IRQ3	11	Serial Channel 1
Master	IRQ4	12	Serial Channel 0
Master	IRQ5	13	LPT2/Spare
Master	IRQ6	14	Floppy Disk
Master	IRQ7	15	LPT1

Table B-3 Interrupt Assignments

Reserved interrupts can be assigned to the PCI interrupts.

LPT2 port interrupt is connected but only one printer port is available for use.

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

B.2.2 PCI Interrupts

There are four PCI interrupts which are active low level sensitive signals.

The interrupts have been connected to the following devices:

PIRQ0	(A)	PCI Expansion, Ethernet, PMC INTD
PIRQ1	(B)	PCI Expansion, PMC INTA
PIRQ2	(C)	SCSI 53C875, PMC INTB, PCI Expansion
PIRQ3	(D)	Universe LINT0, PMC INTC, PCI Expansion

These are wired OR interrupt lines.

B.2.3 NMI

The processor's NMI signal can be generated from the following sources: SERR (PCI), Universe LINT1, or watchdog time-out. The SERR source is generated via the PIIX4E the others are controlled via the status / control registers.

LINT1 is connected directly and uses the Universe registers to control its operation.

SERR is controlled via the PIIX4E.

Watchdog Time-out is controlled via the watchdog register.

LINT1, watchdog time-out and front panel switch connect to NMI via IOCHK# on the PIIX4E and thus IOCHK# must be enabled in the PIIX4E to allow NMI generation by these sources.

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

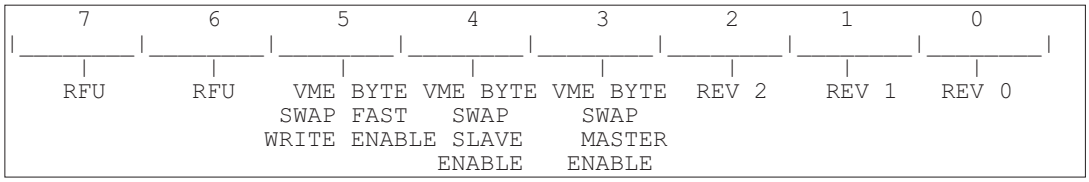
B.2.4 Onboard Status & Control Registers

There are 4 byte wide control and status registers. These are accessed in I/O address space at the following addresses:-

- 210h for Status & Control register 0
- 212h for Status & Control register 1
- 214h for Watchdog Status & Control register
- 216h for Status & Control register 2.
- 217h for Status register 3.

These registers can only be accessed when $\overline{\text{PCS0}}$ on the PIIX4E has been programmed as a peripheral chip select. The factory-fitted firmware will program this bit to enable access by user software.

B.2.4.1 Status & Control Register 0



Bit 2-0, indicates the hardware revision strapping (Read Only)

000 = Rev A,
001 = Rev B etc…

Bit 3 enables VME byte swapping when the VP PSE/P3x is the VME bus master

0 = Swapping disabled,
1 = Swapping enabled

Bit 4 enables VME byte swapping when the VP PSE/P3x is the VME bus slave

0 = Swapping disabled,
1 = Swapping enabled

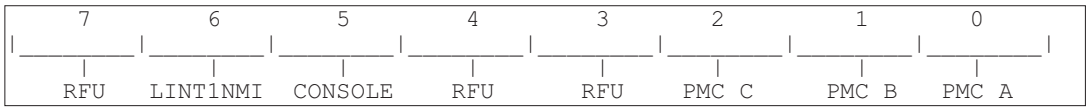
Bit 5 enables fast write swap cycles for both master and slave

When this bit is set to 1 the user must guarantee that write cycles are swappable (D16 or D32 aligned) since the hardware does not perform a full decode

0 = delay enabled,
1 = delay disabled

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

B.2.4.2 Status & Control Register 1



Bits 2-0, indicates the PMC mode 1 status of PMC modules (Read Only)

0 = PCI compliant module not fitted,
1 = PCI compliant module fitted.

PMC A and PMC B are on the AD SBC/PMC PMC carrier board. PMC C is the onboard PMC site.

Bit 4-3, reserved (Read Only)
0 = not fitted,
1 = fitted.

The BIOS will use these bits for factory test operations.

Bit 5, indicates status of console jumper (Read Only)

0 = use COM 1,
1 = use VGA Graphics and keyboard.

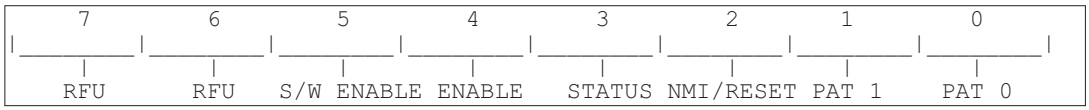
This bit is used by the BIOS to set the standard input/output interface.

Bit 6, indicates whether LINT1 from the Universe is the cause of NMI (Read Only)

0 = has not occurred,
1 = has occurred.

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

B.2.4.3 Watchdog Status & Control Register



Bits 1-0, these bits are used to restart the watchdog timer(Read/Write)

See the section B.2.4.3.1 for a description of how to use the watchdog.

Bit 2, select watchdog action (Read/Write)

This bit selects the following actions when the watchdog times out:-

0 = generate an NMI (default)
1 = generate a board reset.

Bit 3, indicates the status of the watchdog (Read Only)

0 = watchdog timed out
1 = watchdog OK.

This bit can be used to determine if the source of an NMI was due to the watchdog.

Bit 4, indicates the state of the watchdog enable jumper (Read Only)

0 = disable watchdog (default)
1 = enable watchdog

Bit 5, software enable (Read/Write)

0 = disable watchdog(default)
1 = enable watchdog

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

B.2.4.3.1 Watchdog Configuration and Use

The watchdog circuitry contains features to safeguard against accidental use through faulty or unintended software actions. To enable the watchdog the following sequence of events needs to be performed:-

1) read the watchdog register. Check the status of the watchdog enable link (bit 4). If it reads 'low' then proceed to step 2. If it reads 'high' then the watchdog cannot be enabled in software.

2) Set bits 1 & 0 to the complement of each other (0,1 or 1,0) and at the same time set bit 5 'high'.

3) Write the new value back.

4) Complement bits 1 & 0. Write the new value to the watchdog register.

5) Repeat step 4

Once the watchdog has been enabled it can be disabled by repeating the above procedure with bit 5 set 'low'.

To prevent the watchdog timing out it must be restarted at regular intervals. The maximum interval is pre-set to 1 second. This is a function of the watchdog chip and cannot be changed. If a longer period elapses between restarts then the watchdog will time out and cause a reset or NMI depending on the state of bit 2 of the watchdog register. To restart the watchdog the complement of the lower two bits must be written into the watchdog register. These two bits must also be the complement of each other i.e. 0,1 or 1,0. Writing any other value or the same value will not restart the watchdog.

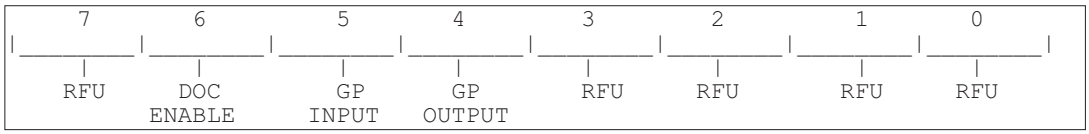
If the watchdog timeout is configured to generate a board reset and a time-out occurs the watchdog circuit will also be reset. The watchdog will therefore need to be re-enabled after a reset has occurred. This has been done to allow operating systems or other software to boot after a reset without having to keep the watchdog from timing out during this period. An NMI will not clear the watchdog, this will have to be done as part of the NMI handler routine. The reason for this is to preserve the status of the watchdog circuit to allow the code to determine the source of the NMI.

NOTE Once an erroneous value has been written into the watchdog register it will take two further writes using the correct values to restart the watchdog again. Therefore, when changing any bit in the register when the watchdog has been enabled, bits 1 & 0 should be complemented.

NOTE The actual time-out of the watchdog chip can vary between 1s and 2.25s. To guarantee correct operation on all boards the restart interval should be less than 1s.

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

B.2.4.4 Status & Control Register 2



Bit 5-4, General Purpose Input & Output

These bits can be used as a general purpose input/output port. They are connected to the P2 connector pins Z3 and Z1. The value written into bit 4 will appear as a TTL logic signal on pin Z1, and the logic level on pin Z3 can be read via bit 5.

Bit 6, DiskOnChip enable

This bit enables a DiskOnChip device when set to 1. When reset to 0 the DiskOnChip is disabled.

B.2.4.5 Status Register 3



Bits 4-0, indicate the VME geographical addresses

If the board is inserted into a VME 64 with extensions backplane, the value read is the inverse of the slot number.

Bit 5, indicates the VME geographical address parity bit

If the board is inserted into a VME 64 with extensions backplane, this bit will be the inverse of the GAP signal logic level.

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

B.3 P.O.S.T. LED

The P.O.S.T. LED is controlled via the speaker port. The P.O.S.T. LED replaces a P.C. speaker and is programmed in the same way a speaker would be programmed.

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

AD VP2/001-10 Breakout

C.1 Introduction

The AD VP2/001-10 product is a 96-way P2 backplane breakout for a number of Concurrent Technologies VME boards. It provides a 50-way IDC SCSI header, a 34-way IDC Floppy header, a 26-way printer header, a 3-way External Reset/NMI header, a 3-way 5V header and additional power connection terminals.

This breakout requires one slot width behind the backplane.

C.2 Layout

Figure C-1 shows the position of connectors and headers.

The AD VP2/001-10 requires a minimum of 70mm depth behind the VME backplane. This measurement is taken from the mating face of the rear P2 connector.

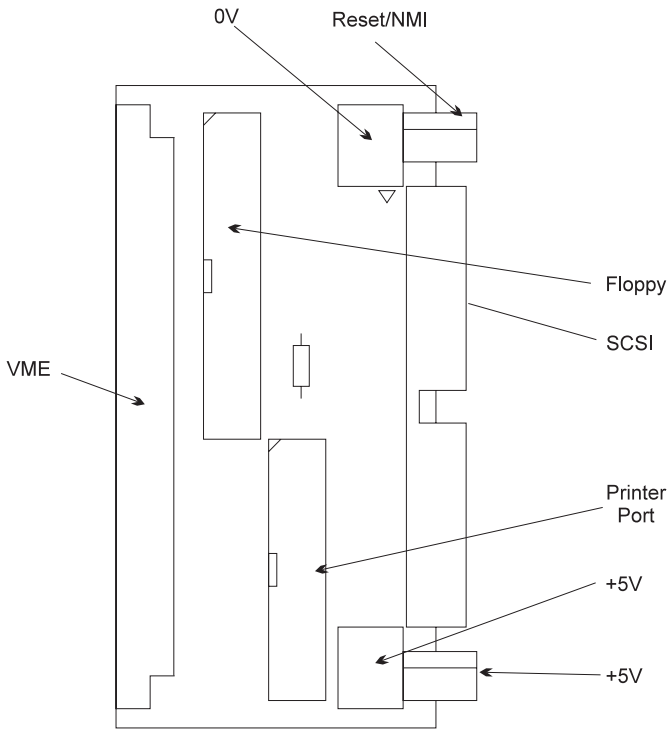


Figure C-1 AD VP2/001-10 P2 Breakout Connectors

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

C.3 Header Configuration

The headers are designed to enable use of standard P.C. Floppy disk and SCSI IDC cables. Detailed below are the Pin-outs of the connector headers.

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	2	$\overline{DBO}$
3	Ground (0V)	4	$\overline{DB1}$
5	Ground (0V)	6	$\overline{DB2}$
7	Ground (0V)	8	$\overline{DB3}$
9	Ground (0V)	10	$\overline{DB4}$
11	Ground (0V)	12	$\overline{DB5}$
13	Ground (0V)	14	$\overline{DB6}$
15	Ground (0V)	16	$\overline{DB7}$
17	Ground (0V)	18	$\overline{DBP}$
19	Ground (0V)	20	Ground (0V)
21	Ground (0V)	22	Ground (0V)
23	Ground (0V)	24	NC
25	Ground (0V)	26	TERMPWR
27	Ground (0V)	28	NC
29	Ground (0V)	30	Ground (0V)
31	Ground (0V)	32	$\overline{ATN}$
33	Ground (0V)	34	Ground (0V)
35	Ground (0V)	36	$\overline{BSY}$
37	Ground (0V)	38	$\overline{ACK}$
39	Ground (0V)	40	$\overline{RST}$
41	Ground (0V)	42	$\overline{MSG}$
43	Ground (0V)	44	$\overline{SEL}$
45	Ground (0V)	46	$\overline{CD}$
47	Ground (0V)	48	$\overline{REQ}$
49	Ground (0V)	50	$\overline{SIO}$

Table C-1 SCSI 50-way IDC Header

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	2	DRVDEN0
3	Ground (0V)	4	NC
5	Ground (0V)	6	DRVDEN1
7	Ground (0V)	8	$\overline{INDX}$
9	Ground (0V)	10	$\overline{FDME0}$
11	Ground (0V)	12	$\overline{FDSI}$
13	Ground (0V)	14	$\overline{FDS0}$
15	Ground (0V)	16	$\overline{FDME1}$
17	Ground (0V)	18	$\overline{DIR}$
19	Ground (0V)	20	$\overline{STEP}$
21	Ground (0V)	22	$\overline{WRDATA}$
23	Ground (0V)	24	$\overline{WE}$
25	Ground (0V)	26	$\overline{TRK0}$
27	Ground (0V)	28	$\overline{WP}$
29	Ground (0V)	30	$\overline{RDDATA}$
31	Ground (0V)	32	$\overline{HDSEL}$
33	Ground (0V)	34	$\overline{DSKCHG}$

Table C-2 Floppy 34-way IDC Header

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

Pin No.	Signal	Pin No.	Signal
1	$\overline{\text{STROBE}}$	2	$\overline{\text{AFD}}$
3	PRT D0	4	ERR
5	PRT D1	6	$\overline{\text{INIT}}$
7	PRT D2	8	$\overline{\text{SELECT I}}$
9	PRT D3	10	Ground (0V)
11	PRT D4	12	Ground (0V)
13	PRT D5	14	Ground (0V)
15	PRT D6	16	Ground (0V)
17	PRT D7	18	Ground (0V)
19	$\overline{\text{PRTACK}}$	20	Ground (0V)
21	PRT BUSY	22	Ground (0V)
23	PERR	24	Ground (0V)
25	SELECT	26	Ground (0V)

Table C-3 Printer 26-way IDC Header

Pin No.	Signal
1	Ground (0V)
2	Reset / NMI
3	Ground (0V)

Table C-4 External Reset/NMI 3-way Header

Pin No.	Signal
1	+5V
2	+5V
3	+5V

Table C-5 5V 3-way Header

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

This page has been left intentionally blank.

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

AD VP2/001-20 Breakout

D.1 Introduction

The AD VP2/001-20 product is a 96-way P2 backplane breakout for a number of Concurrent Technologies VME boards. It provides a 40-way IDC EIDE header, a 34-way IDC Floppy header, a 26-way printer header, a 3-way External Reset/NMI header, a 3-way 5V header and additional power connection terminals.

This breakout requires one slot width behind the backplane.

D.2 Layout

Figure D-1 shows the position of connectors and headers.

The AD VP2/001-20 requires a minimum of 70mm depth behind the VME backplane. This measurement is taken from the mating face of the rear P2 connector.

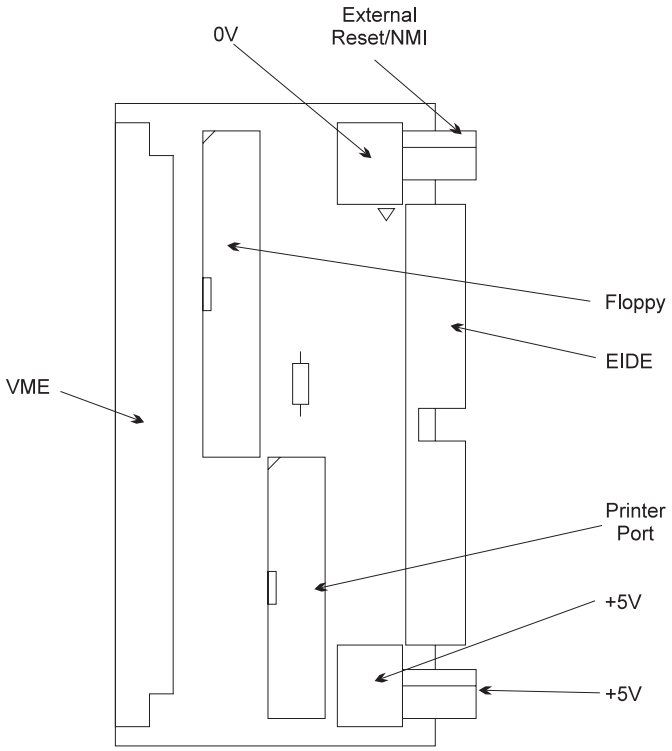


Figure D-1 AD VP2/001-20 P2 Breakout Connectors

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

D.3 Header Configuration

The headers are designed to enable use of standard P.C. Floppy disk and EIDE IDC cables.

Detailed below are the pin-outs of the connector headers.

Pin No.	Signal	Pin No.	Signal
1	$\overline{\text{RST}}$	2	Ground (0V)
3	D7	4	D8
5	D6	6	D9
7	D5	8	D10
9	D4	10	D11
11	D3	12	D12
13	D2	14	D13
15	D1	16	D14
17	D0	18	D15
19	Ground (0V)	20	NC
21	DRQ	22	Ground (0V)
23	$\overline{\text{IOWA}}$	24	Ground (0V)
25	$\overline{\text{IORA}}$	26	Ground (0V)
27	RDY	28	NC
29	$\overline{\text{ACK}}$	30	Ground (0V)
31	IRQ	32	NC
33	A1	34	NC
35	A0	36	A2
37	$\overline{\text{CS1}}$	38	$\overline{\text{CS2}}$
39	$\overline{\text{ASP}}$	40	Ground (0V)

Table D-1 EIDE 40-way IDC Header

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	2	DRVDEN0
3	Ground (0V)	4	NC
5	Ground (0V)	6	DRVDEN1
7	Ground (0V)	8	$\overline{\text{INDX}}$
9	Ground (0V)	10	$\overline{\text{FDME0}}$
11	Ground (0V)	12	$\overline{\text{FDSI}}$
13	Ground (0V)	14	$\overline{\text{FDS0}}$
15	Ground (0V)	16	$\overline{\text{FDME1}}$
17	Ground (0V)	18	$\overline{\text{DIR}}$
19	Ground (0V)	20	$\overline{\text{STEP}}$
21	Ground (0V)	22	$\overline{\text{WRDATA}}$
23	Ground (0V)	24	$\overline{\text{WE}}$
25	Ground (0V)	26	$\overline{\text{TRK0}}$
27	Ground (0V)	28	$\overline{\text{WP}}$
29	Ground (0V)	30	$\overline{\text{RDDATA}}$
31	Ground (0V)	32	$\overline{\text{HDSEL}}$
33	Ground (0V)	34	$\overline{\text{DSKCHG}}$

Table D-2 Floppy 34-way IDC Header

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

Pin No.	Signal	Pin No.	Signal
1	$\overline{\text{STROBE}}$	2	$\overline{\text{AFD}}$
3	PRT D0	4	ERR
5	PRT D1	6	$\overline{\text{INIT}}$
7	PRT D2	8	$\overline{\text{SELECT I}}$
9	PRT D3	10	Ground (0V)
11	PRT D4	12	Ground (0V)
13	PRT D5	14	Ground (0V)
15	PRT D6	16	Ground (0V)
17	PRT D7	18	Ground (0V)
19	$\overline{\text{PRTACK}}$	20	Ground (0V)
21	PRT BUSY	22	Ground (0V)
23	PERR	24	Ground (0V)
25	SELECT	26	Ground (0V)

Table D-3 Printer 26-way IDC Header

Pin No.	Signal
1	Ground (0V)
2	Reset / NMI
3	Ground (0V)

Table D-4 External Reset/NMI 3-way Header

Pin No.	Signal
1	+5V
2	+5V
3	+5V

Table D-5 5V 3-way Header

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

This page has been left intentionally blank.

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

AD VP2/002-10 Breakout

E.1 Introduction

The AD VP2/002-10 product is a 160-way P2 backplane breakout for a number of Concurrent Technologies VME boards. It provides a 40-way IDC EIDE header, a 50-way IDC SCSI header,a 34-way IDC Floppy header, a USB connector, a 26-way printer header, a 3-way External Reset/NMI header, a 3way 5V header and additional power connection terminals.

This breakout requires one slot width behind the backplane.

E.2 Layout

Figure E-1 shows the position of connectors and headers.

The AD VP2/002-10 requires a minimum of 82mm depth behind the VME backplane. This measurement is taken from the mating face of the rear P2 connector.

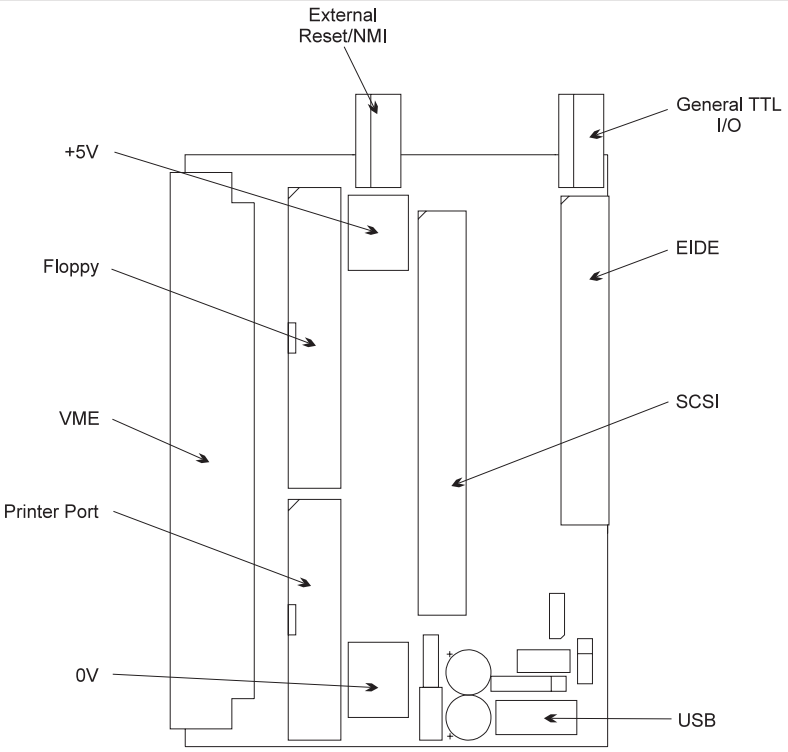


Figure E-1 AD VP2/002-10 P2 Breakout Connectors

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

E.3 Header Configuration

The headers are designed to enable use of standard P.C. Floppy disk, SCSI IDC and EIDE (ATA) cables. Detailed below are the Pin-outs of the connector headers.

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	2	$\overline{DBO}$
3	Ground (0V)	4	$\overline{DB1}$
5	Ground (0V)	6	$\overline{DB2}$
7	Ground (0V)	8	$\overline{DB3}$
9	Ground (0V)	10	$\overline{DB4}$
11	Ground (0V)	12	$\overline{DB5}$
13	Ground (0V)	14	$\overline{DB6}$
15	Ground (0V)	16	$\overline{DB7}$
17	Ground (0V)	18	$\overline{DBP}$
19	Ground (0V)	20	Ground (0V)
21	Ground (0V)	22	Ground (0V)
23	Ground (0V)	24	NC
25	Ground (0V)	26	TERMPWR
27	Ground (0V)	28	NC
29	Ground (0V)	30	Ground (0V)
31	Ground (0V)	32	$\overline{ATN}$
33	Ground (0V)	34	Ground (0V)
35	Ground (0V)	36	$\overline{BSY}$
37	Ground (0V)	38	$\overline{ACK}$
39	Ground (0V)	40	$\overline{RST}$
41	Ground (0V)	42	$\overline{MSG}$
43	Ground (0V)	44	$\overline{SEL}$
45	Ground (0V)	46	$\overline{CD}$
47	Ground (0V)	48	$\overline{REQ}$
49	Ground (0V)	50	$\overline{SIO}$

Table E-1 SCSI 50-way IDC Header

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

Pin No.	Signal	Pin No.	Signal
1	$\overline{RST}$	2	Ground (0V)
3	D7	4	D8
5	D6	6	D9
7	D5	8	D10
9	D4	10	D11
11	D3	12	D12
13	D2	14	D13
15	D1	16	D14
17	D0	18	D15
19	Ground (0V)	20	NC
21	DRQ	22	Ground (0V)
23	$\overline{IOWA}$	24	Ground (0V)
25	$\overline{IORA}$	26	Ground (0V)
27	RDY	28	NC
29	$\overline{ACK}$	30	Ground (0V)
31	IRQ	32	NC
33	A1	34	NC
35	A0	36	A2
37	$\overline{CS1}$	38	$\overline{CS2}$
39	$\overline{ASP}$	40	Ground (0V)

Table E-2 EIDE 40-way IDC Header

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	2	DRVDEN0
3	Ground (0V)	4	NC
5	Ground (0V)	6	DRVDEN1
7	Ground (0V)	8	$\overline{INDX}$
9	Ground (0V)	10	$\overline{FDME0}$
11	Ground (0V)	12	$\overline{FDSI}$
13	Ground (0V)	14	$\overline{FDS0}$
15	Ground (0V)	16	$\overline{FDME1}$
17	Ground (0V)	18	$\overline{DIR}$
19	Ground (0V)	20	$\overline{STEP}$
21	Ground (0V)	22	$\overline{WRDATA}$
23	Ground (0V)	24	$\overline{WE}$
25	Ground (0V)	26	$\overline{TRK0}$
27	Ground (0V)	28	$\overline{WP}$
29	Ground (0V)	30	$\overline{RDDATA}$
31	Ground (0V)	32	$\overline{HDSEL}$
33	Ground (0V)	34	$\overline{DSKCHG}$

Table E-3 Floppy 34-way IDC Header

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

Pin No.	Signal	Pin No.	Signal
1	$\overline{\text{STROBE}}$	2	$\overline{\text{AFD}}$
3	PRT D0	4	$\overline{\text{ERR}}$
5	PRT D1	6	$\overline{\text{INIT}}$
7	PRT D2	8	$\overline{\text{SELECT I}}$
9	PRT D3	10	Ground (0V)
11	PRT D4	12	Ground (0V)
13	PRT D5	14	Ground (0V)
15	PRT D6	16	Ground (0V)
17	PRT D7	18	Ground (0V)
19	$\overline{\text{PRTACK}}$	20	Ground (0V)
21	PRT BUSY	22	Ground (0V)
23	PERR	24	Ground (0V)
25	SELECT	26	Ground (0V)

Table E-4 Printer 26-way IDC Header

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

Pin No.	Signal
1	USB Vcc (+5V)
2	USB $\overline{\text{Data}}$
3	USB Data
4	USB Ground
SCR	SCREEN (SHIELD)

Table E-5 USB Connector

NOTE USB SCREEN is connected to Ground (0V) via a 0.1µF Capacitor

Pin No.	Signal
1	USB Vcc (+5V)
2	USB $\overline{\text{Data}}$
3	USB Data
4	USB Ground
5	SCREEN (SHIELD)

Table E-6 USB 5-pin Header

Pin No.	Signal
1	Ground (0V)
2	Reset / NMI
3	Ground (0V)

Table E-7 External Reset/NMI 3-way Header

Pin No.	Signal
1	+5V
2	+5V
3	+5V

Table E-8 5V 3-way Header

Pin No.	Signal
1	General purpose output
2	Ground (0V)
3	General purpose input

Table E-9 General purpose I/O Header

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

This page has been left intentionally blank

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

AD VP2/003-10 Breakout

F.1 Introduction

The AD VP2/003-10 product is a 160-way P2 backplane breakout for the VP PSE/P3x. It provides a 68-way right angled female wide SCSI connector, a 40-way IDC EIDE header, a 20-way right angled DFP connector, a 50-way IDC Narrow SCSI header, a 34-way IDC Floppy header, a 4-way right angled USB connector, a 3-way External Reset header, and a 3-way general purpose I/O header.

This breakout requires one slot width behind the backplane.

F.2 Layout

Figure F-1 shows the position of connectors and headers.

The AD VP2/003-10 requires a minimum of 90mm depth behind the VME backplane.

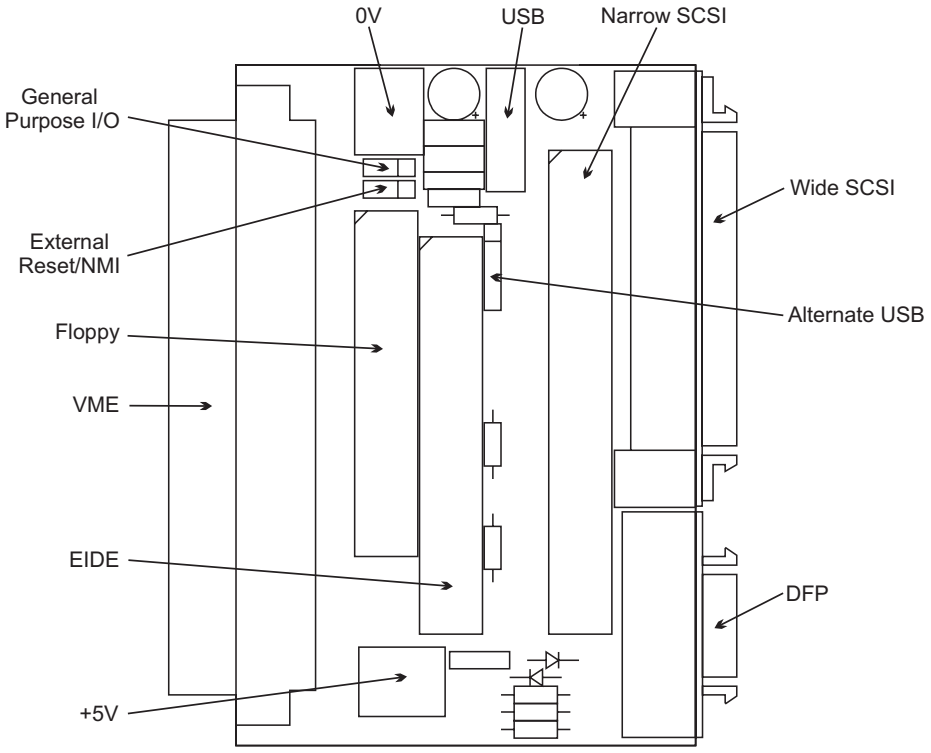


Figure F-1 AD VP2/003-10 P2 Breakout Connectors

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

F.3 Header and Connector Configuration

The headers are designed to enable use of standard P.C. Floppy disk, SCSI IDC and EIDE (ATA) cables. Detailed below are the Pin-outs of the connector headers.

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	2	$\overline{DB0}$
3	Ground (0V)	4	$\overline{DB1}$
5	Ground (0V)	6	$\overline{DB2}$
7	Ground (0V)	8	$\overline{DB3}$
9	Ground (0V)	10	$\overline{DB4}$
11	Ground (0V)	12	$\overline{DB5}$
13	Ground (0V)	14	$\overline{DB6}$
15	Ground (0V)	16	$\overline{DB7}$
17	Ground (0V)	18	$\overline{DBP}$
19	Ground (0V)	20	Ground (0V)
21	Ground (0V)	22	Ground (0V)
23	Ground (0V)	24	NC
25	Ground (0V)	26	TERMPWR
27	Ground (0V)	28	NC
29	Ground (0V)	30	Ground (0V)
31	Ground (0V)	32	$\overline{ATN}$
33	Ground (0V)	34	Ground (0V)
35	Ground (0V)	36	$\overline{BSY}$
37	Ground (0V)	38	$\overline{ACK}$
39	Ground (0V)	40	$\overline{RST}$
41	Ground (0V)	42	$\overline{MSG}$
43	Ground (0V)	44	$\overline{SEL}$
45	Ground (0V)	46	$\overline{CD}$
47	Ground (0V)	48	$\overline{REQ}$
49	Ground (0V)	50	$\overline{SIO}$

Table F-1 Narrow SCSI 50-way IDC Header

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

Pin No.	Signal	Pin No.	Signal
1	$\overline{\text{RST}}$	2	Ground (0V)
3	D7	4	D8
5	D6	6	D9
7	D5	8	D10
9	D4	10	D11
11	D3	12	D12
13	D2	14	D13
15	D1	16	D14
17	D0	18	D15
19	Ground (0V)	20	NC
21	DRQ	22	Ground (0V)
23	$\overline{\text{IOWA}}$	24	Ground (0V)
25	$\overline{\text{IORA}}$	26	Ground (0V)
27	RDY	28	NC
29	$\overline{\text{ACK}}$	30	Ground (0V)
31	IRQ	32	NC
33	A1	34	NC
35	A0	36	A2
37	$\overline{\text{CS1}}$	38	$\overline{\text{CS2}}$
39	$\overline{\text{ASP}}$	40	Ground (0V)

Table F-2 EIDE 40-way IDC Header

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	2	DRVDEN0
3	Ground (0V)	4	NC
5	Ground (0V)	6	DRVDEN1
7	Ground (0V)	8	$\overline{\text{INDX}}$
9	Ground (0V)	10	$\overline{\text{FDME0}}$
11	Ground (0V)	12	$\overline{\text{FDSI}}$
13	Ground (0V)	14	$\overline{\text{FDS0}}$
15	Ground (0V)	16	$\overline{\text{FDME1}}$
17	Ground (0V)	18	$\overline{\text{DIR}}$
19	Ground (0V)	20	$\overline{\text{STEP}}$
21	Ground (0V)	22	$\overline{\text{WRDATA}}$
23	Ground (0V)	24	$\overline{\text{WE}}$
25	Ground (0V)	26	$\overline{\text{TRK0}}$
27	Ground (0V)	28	$\overline{\text{WP}}$
29	Ground (0V)	30	$\overline{\text{RDDATA}}$
31	Ground (0V)	32	$\overline{\text{HDSEL}}$
33	Ground (0V)	34	$\overline{\text{DSKCHG}}$

Table F-3 Floppy 34-way IDC Header

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

Pin No.	Signal
1	USB Vcc (+5V)
2	USB $\overline{\text{Data}}$
3	USB Data
4	USB Ground
SCR	SCREEN (SHIELD)

Table F-4 USB Connector

NOTE USB SCREEN is connected to Ground (0V) via a 0.1µF Capacitor

Pin No.	Signal
1	USB Vcc (+5V)
2	USB $\overline{\text{Data}}$
3	USB Data
4	USB Ground
SCR	SCREEN (SHIELD)

Table F-5 Alternate USB Header

Pin No.	Signal
1	Ground (0V)
2	Reset / NMI
3	Ground (0V)

Table F-6 External Reset/NMI 3-way Header

Pin No.	Signal
1	GP_OUT
2	Ground (0V)
3	GP_IN

Table F-7 General Purpose I/O Header

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

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	35	DB12*
2	Ground (0V)	36	DB13*
3	Ground (0V)	37	DB14*
4	Ground (0V)	38	DB15*
5	Ground (0V)	39	DBP1*
6	Ground (0V)	40	DB0*
7	Ground (0V)	41	DB1*
8	Ground (0V)	42	DB2*
9	Ground (0V)	43	DB3*
10	Ground (0V)	44	DB4*
11	Ground (0V)	45	DB5*
12	Ground (0V)	46	DB6*
13	Ground (0V)	47	DB7*
14	Ground (0V)	48	DBP0*
15	Ground (0V)	49	Ground (0V)
16	Ground (0V)	50	Ground (0V)
17	TERMPWR	51	TERMPWR
18	TERMPWR	52	TERMPWR
19	NC	53	NC
20	Ground (0V)	54	Ground (0V)
21	Ground (0V)	55	ATN*
22	Ground (0V)	56	Ground (0V)
23	Ground (0V)	57	BSY*
24	Ground (0V)	58	ACK*
25	Ground (0V)	59	RST*
26	Ground (0V	)60	MSG
27	Ground (0V)	61	SEL*
28	Ground (0V)	62	CD*
29	Ground (0V)	63	REQ*
30	Ground (0V)	64	IO*
31	Ground (0V)	65	DB8*
32	Ground (0V)	66	DB9*
33	Ground (0V)	67	DB10
34	Ground (0V)	68	DB11*

* Denotes active low signals

Table F-8 Wide SCSI 68-way Connector

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

Pin No.	Signal	Pin No.	Signal
1	TX1	2	$\overline{TX1}$
3	Ground(0V)	4	Ground(0V)
5	TXC	6	$\overline{TXC}$
7	Ground(0V)	8	DFP VCC (+5V)
9	NC	10	NC
11	TX2	12	$\overline{TX2}$
13	Ground(0V)	14	Ground(0V)
15	TX0	16	$\overline{TX0}$
17	NC	18	HPD
19	GDC DAT	20	GDCCK

Table F-9 DFP Connector

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

AD VP2/003-20 Breakout

G.1 Introduction

The AD VP2/003-20 product is a 96-way P2 backplane breakout for the VP PSE/P3x. It provides a 40-way IDC EIDE header, a 20-way right angled DFP connector, and a 34-way IDC Floppy header.

This breakout requires one slot width behind the backplane.

G.2 Layout

Figure F-1 shows the position of connectors and headers.

The AD VP2/003-20 requires a minimum of 90mm depth behind the VME backplane.

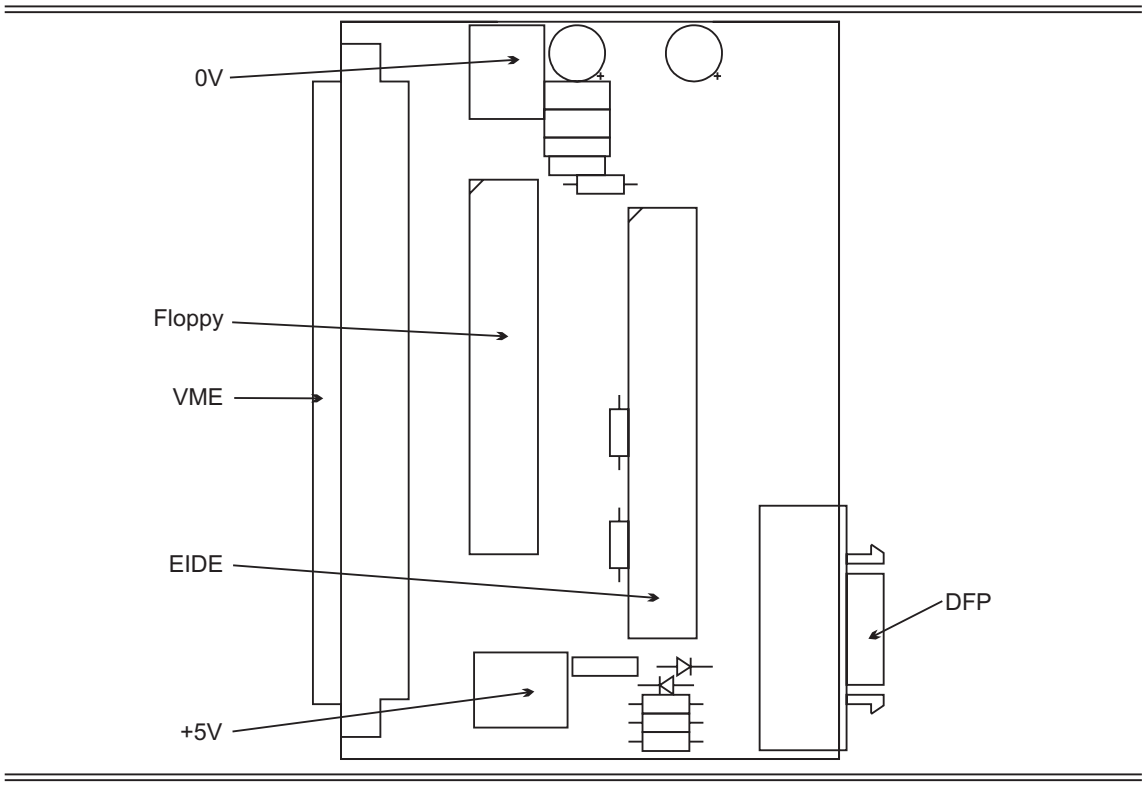


Figure G-1 AD VP2/003-20 P2 Breakout Connectors

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

G.3 Header and Connector Configuration

The headers are designed to enable use of standard P.C. Floppy disk and EIDE (ATA) cables. Detailed below are the Pin-outs of the connector headers.

Pin No.	Signal	Pin No.	Signal
1	$\overline{\text{RST}}$	2	Ground (0V)
3	D7	4	D8
5	D6	6	D9
7	D5	8	D10
9	D4	10	D11
11	D3	12	D12
13	D2	14	D13
15	D1	16	D14
17	D0	18	D15
19	Ground (0V)	20	NC
21	DRQ	22	Ground (0V)
23	$\overline{\text{IOWA}}$	24	Ground (0V)
25	$\overline{\text{IORA}}$	26	Ground (0V)
27	RDY	28	NC
29	$\overline{\text{ACK}}$	30	Ground (0V)
31	IRQ	32	NC
33	A1	34	NC
35	A0	36	A2
37	$\overline{\text{CS1}}$	38	$\overline{\text{CS2}}$
39	$\overline{\text{ASP}}$	40	Ground (0V)

Table G-1 EIDE 40-way IDC Header

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	2	DRVDEN0
3	Ground (0V)	4	NC
5	Ground (0V)	6	DRVDEN1
7	Ground (0V)	8	$\overline{\text{INDX}}$
9	Ground (0V)	10	$\overline{\text{FDME0}}$
11	Ground (0V)	12	$\overline{\text{FDSI}}$
13	Ground (0V)	14	$\overline{\text{FDS0}}$
15	Ground (0V)	16	$\overline{\text{FDME1}}$
17	Ground (0V)	18	$\overline{\text{DIR}}$
19	Ground (0V)	20	$\overline{\text{STEP}}$
21	Ground (0V)	22	$\overline{\text{WRDATA}}$
23	Ground (0V)	24	$\overline{\text{WE}}$
25	Ground (0V)	26	$\overline{\text{TRK0}}$
27	Ground (0V)	28	$\overline{\text{WP}}$
29	Ground (0V)	30	$\overline{\text{RDDATA}}$
31	Ground (0V)	32	$\overline{\text{HDSEL}}$
33	Ground (0V)	34	$\overline{\text{DSKCHG}}$

Table G-2 Floppy 34-way IDC Header

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

Pin No.	Signal	Pin No.	Signal
1	TX1	2	$\overline{TX1}$
3	Ground(0V)	4	Ground(0V)
5	TXC	6	$\overline{TXC}$
7	Ground(0V)	8	DFP VCC (+5V)
9	NC	10	NC
11	TX2	12	$\overline{TX2}$
13	Ground(0V)	14	Ground(0V)
15	TX0	16	$\overline{TX0}$
17	NC	18	Reserved
19	NC	20	NC

Table G-3 DFP Connector

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

This page has been left intentionally blank

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

AD VP2/003-30 Breakout

H.1 Introduction

The AD VP2/003-30 product is a 96-way P2 backplane breakout for the VP PSE/P3x. It provides a 68-way right angled female wide SCSI connector, a 20-way right angled DFP connector, a 50-way IDC Narrow SCSI header, a 34-way IDC Floppy header, and a 3-way External Reset header.

This breakout requires one slot width behind the backplane.

H.2 Layout

Figure F-1 shows the position of connectors and headers.

The AD VP2/003-30 requires a minimum of 90mm depth behind the VME backplane.

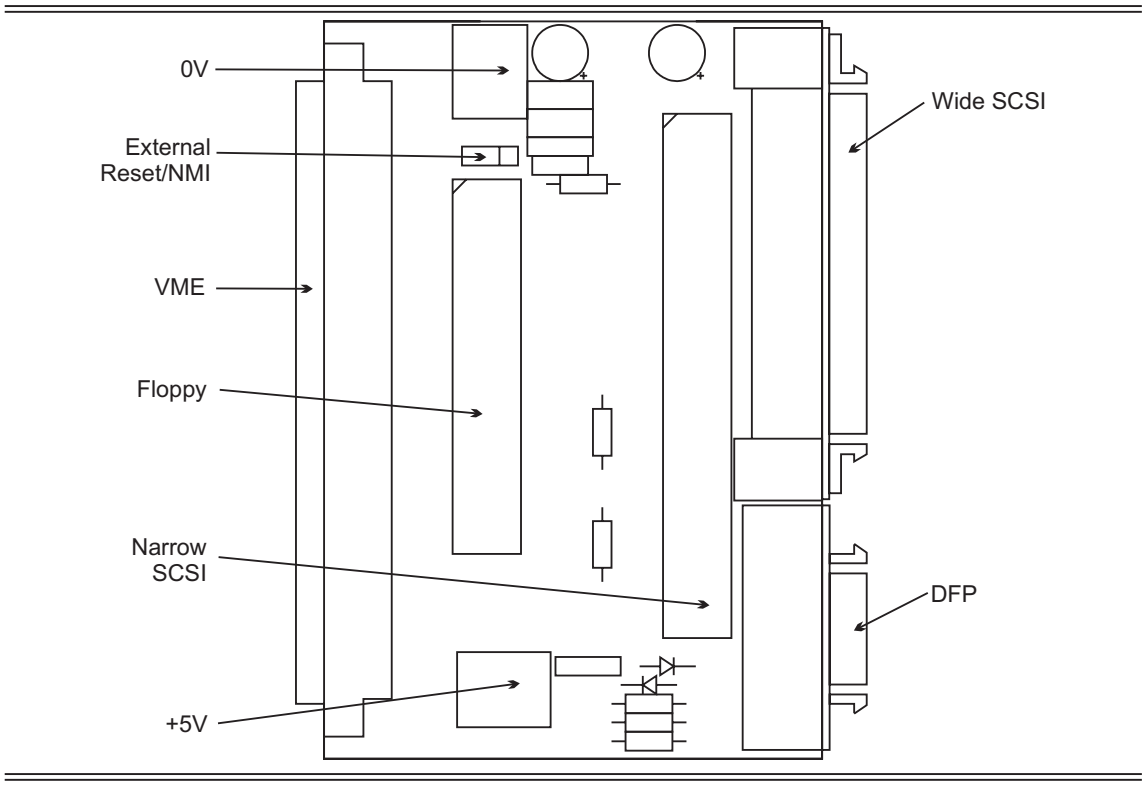


Figure H-1 AD VP2/003-30 P2 Breakout Connectors

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

H.3 Header and Connector Configuration

The headers are designed to enable use of standard P.C. Floppy disk, SCSI IDC and EIDE (ATA) cables. Detailed below are the Pin-outs of the connector headers.

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	2	$\overline{DB0}$
3	Ground (0V)	4	$\overline{DB1}$
5	Ground (0V)	6	$\overline{DB2}$
7	Ground (0V)	8	$\overline{DB3}$
9	Ground (0V)	10	$\overline{DB4}$
11	Ground (0V)	12	$\overline{DB5}$
13	Ground (0V)	14	$\overline{DB6}$
15	Ground (0V)	16	$\overline{DB7}$
17	Ground (0V)	18	$\overline{DBP}$
19	Ground (0V)	20	Ground (0V)
21	Ground (0V)	22	Ground (0V)
23	Ground (0V)	24	NC
25	Ground (0V)	26	TERMPWR
27	Ground (0V)	28	NC
29	Ground (0V)	30	Ground (0V)
31	Ground (0V)	32	$\overline{ATN}$
33	Ground (0V)	34	Ground (0V)
35	Ground (0V)	36	$\overline{BSY}$
37	Ground (0V)	38	$\overline{ACK}$
39	Ground (0V)	40	$\overline{RST}$
41	Ground (0V)	42	$\overline{MSG}$
43	Ground (0V)	44	$\overline{SEL}$
45	Ground (0V)	46	$\overline{CD}$
47	Ground (0V)	48	$\overline{REQ}$
49	Ground (0V)	50	$\overline{SIO}$

Table H-1 Narrow SCSI 50-way IDC Header

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	2	DRVDEN0
3	Ground (0V)	4	NC
5	Ground (0V)	6	DRVDEN1
7	Ground (0V)	8	$\overline{INDX}$
9	Ground (0V)	10	$\overline{FDME0}$
11	Ground (0V)	12	$\overline{FDSI}$
13	Ground (0V)	14	$\overline{FDS0}$
15	Ground (0V)	16	$\overline{FDME1}$
17	Ground (0V)	18	$\overline{DIR}$
19	Ground (0V)	20	$\overline{STEP}$
21	Ground (0V)	22	$\overline{WRDATA}$
23	Ground (0V)	24	$\overline{WE}$
25	Ground (0V)	26	$\overline{TRK0}$
27	Ground (0V)	28	$\overline{WP}$
29	Ground (0V)	30	$\overline{RDDATA}$
31	Ground (0V)	32	$\overline{HDSEL}$
33	Ground (0V)	34	$\overline{DSKCHG}$

Table H-2 Floppy 34-way IDC Header

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

Pin No.	Signal
1	Ground (0V)
2	Reset / NMI
3	Ground (0V)

Table H-3 External Reset/NMI 3-way Header

Pin No.	Signal	Pin No.	Signal
1	Ground (0V)	35	DB12*
2	Ground (0V)	36	DB13*
3	Ground (0V)	37	DB14*
4	Ground (0V)	38	DB15*
5	Ground (0V)	39	DBP1*
6	Ground (0V)	40	DB0*
7	Ground (0V)	41	DB1*
8	Ground (0V)	42	DB2*
9	Ground (0V)	43	DB3*
10	Ground (0V)	44	DB4*
11	Ground (0V)	45	DB5*
12	Ground (0V)	46	DB6*
13	Ground (0V)	47	DB7*
14	Ground (0V)	48	DBP0*
15	Ground (0V)	49	Ground (0V)
16	Ground (0V)	50	Ground (0V)
17	TERMPWR	51	TERMPWR
18	TERMPWR	52	TERMPWR
19	NC	53	NC
20	Ground (0V)	54	Ground (0V)
21	Ground (0V)	55	ATN*
22	Ground (0V)	56	Ground (0V)
23	Ground (0V)	57	BSY*
24	Ground (0V)	58	ACK*
25	Ground (0V)	59	RST*
26	Ground (0V	)60	MSG
27	Ground (0V)	61	SEL*
28	Ground (0V)	62	CD*
29	Ground (0V)	63	REQ*
30	Ground (0V)	64	IO*
31	Ground (0V)	65	DB8*
32	Ground (0V)	66	DB9*
33	Ground (0V)	67	DB10
34	Ground (0V)	68	DB11*

* Denotes active low signals

Table H-4 Wide SCSI 68-way Connector

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

Pin No.	Signal	Pin No.	Signal
1	TX1	2	$\overline{TX1}$
3	Ground(0V)	4	Ground(0V)
5	TXC	6	$\overline{TXC}$
7	Ground(0V)	8	DFP VCC (+5V)
9	NC	10	NC
11	TX2	12	$\overline{TX2}$
13	Ground(0V)	14	Ground(0V)
15	TX0	16	$\overline{TX0}$
17	NC	18	Reserved
19	NC	20	NC

Table H-5 DFP Connector

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