Artisan Technology Group

# Artisan Technology Group is your source for quality new and certified-used/pre-owned equipment

- FAST SHIPPING AND DELIVERY
- TENS OF THOUSANDS OF IN-STOCK ITEMS
- EQUIPMENT DEMOS
- HUNDREDS OF MANUFACTURERS SUPPORTED
- LEASING/MONTHLY RENTALS
- ITAR CERTIFIED SECURE ASSET SOLUTIONS

**SERVICE CENTER REPAIRS**
Experienced engineers and technicians on staff at our full-service, in-house repair center

**WE BUY USED EQUIPMENT**
Sell your excess, underutilized, and idle used equipment
We also offer credit for buy-backs and trade-ins
***www.artisantg.com/WeBuyEquipment*** ↗

***InstraView***™ **REMOTE INSPECTION**
Remotely inspect equipment before purchasing with our interactive website at ***www.instraview.com*** ↗

**LOOKING FOR MORE INFORMATION?**
Visit us on the web at ***www.artisantg.com*** ↗ for more information on price quotations, drivers, technical specifications, manuals, and documentation

***Contact us:*** (888) 88-SOURCE | sales@artisantg.com | www.artisantg.com

# V5B

## Technical Manual

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

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

# V5B

*Pentium PC/AT Compatible VMEbus Single Board Computer*

## Technical Manual

Copyright © 2001 by SBS Embedded Computers Inc.

*9100-31-039-D1*

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

**V5B Technical Manual**
Document Number **A-935-MN-03765-D1**
Part Number **9100-31-039-D1**
Revision D1

This manual applies to the V5B Embedded PC, revision D1 and above, until superceded.

The information contained within this document has been carefully checked and is believed to be entirely reliable and consistent with the product that it describes. However, no responsibility is assumed for inaccuracies. SBS Embedded Computers Inc. assumes no liability due to the application or use of any product or circuit described herein. SBS Embedded Computers Inc. reserves the right to make changes to any product and product documentation in an effort to improve performance, reliability, or design. Furthermore, the information contained herein is of a proprietary nature and is not to be reproduced without prior written consent of SBS Embedded Computers Inc.

Intel is a trademark of Intel Corporation.

IBM, PC/AT, and AT are trademarks of International Business Machines Corporation.

MS-DOS is a trademark of Microsoft Corporation.

Tundra and Universe are trademarks of Tundra Semiconductor, Inc.

This manual uses some generally accepted conventions for clarity and accuracy. These include:

- The use of an ‘H’ suffix before a number indicates hexadecimal (base sixteen) notation.
- The use of a ‘-’ (minus) suffix to a signal name indicates an active low signal. The signal is either true when it is at a logic zero level or the signal initiates actions on a high-to-low transition.
- Text in `Courier` Font indicates a command entry or output from an SBS embedded PC product using its built-in character set.

This manual is designed to provide information regarding the general use and application of the V5B Embedded PC, as well as detail the hardware design. Software methods and programming information are also provided in Chapters 6 through 10 of this manual.



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

# Contents





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

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

# Tables





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

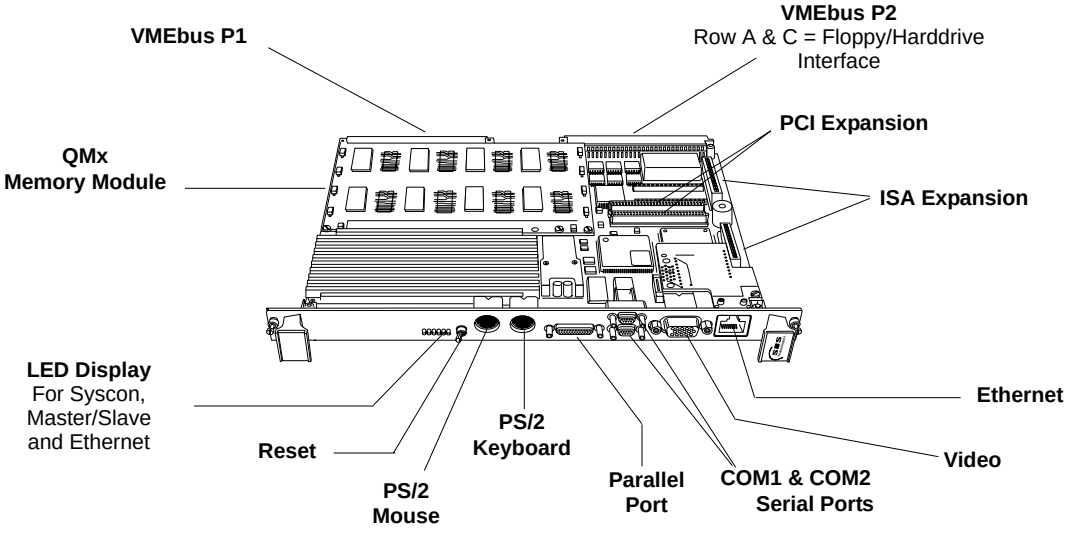
# Figures






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

# CHAPTER 1 Introduction




## Chapter Scope

This chapter describes the features, capabilities and compatibilities of the V5B.

## IBM PC/AT Compatibility Features

The V5B Embedded PC is completely software compatible with the IBM PC/AT personal computer. All of the functions necessary for operation as a PC/AT are included. There are also a number of features provided on board which usually require plug-in cards in a desktop PC/AT. The following paragraphs describe features which relate to the V5B PC/AT functionality.

**Intel Pentium Microprocessors**

The V5B is available from the factory with an Intel Pentium processor at variety of speeds. The 166MHz, 200MHz and 233MHz Pentium MMX processors are currently supported, and support for future Pentium processors is provided. The V5B is also available with the AMD K6-2 microprocessor at speeds of 266MHz, 300MHz and 333MHz. These are x86 compatible. Each processor will run standard software written for the PC/AT. They provide the capability of running 8-bit (8086), 16-bit (80286), as well as full 32-bit (80386) software and operating systems. This choice of processors allows you to select a V5B with the performance best suited to your application.



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

### Compatible ROM BIOS

The V5B uses a ROM BIOS produced by Award BIOS. to ensure that the BIOS is 100% compatible with the IBM PC/AT without infringing on IBM's copyrights. The BIOS provides low-level machine dependent software functions, therefore relieving MS-DOS or any other common operating system from dealing with the hardware directly. Since a great many software programs use calls to the BIOS rather than to MS-DOS, it is important to have a BIOS that functions exactly like the original PC/AT BIOS.

### Compatible I/O

The V5B incorporates the I/O capabilities of a complete PC/AT by using the SMC FDC37C665 super I/O controller. These capabilities include two RS-232 serial communications ports, a parallel printer port, and a keyboard port. Also, a serial mouse port is provided. The COM-2 serial port is enhanced for selectable RS-422 or RS-485 operation.

### System Memory

The V5B is available from the factory with EDO RAM memory technology with a variety of sizes.

### Floppy Controller

The SMC FDC37C665 also includes a fully PC/AT compatible floppy disk controller onboard the V5B. It can control two separate 3.5" 1.44M floppy disk drives. The floppy control signals are provided through the P2 connector for simple connection to an SBS Embedded Computers VME-6200 series disk drive module, or to an external disk drive setup using SBS Embedded Computers' VME-TB21 transition board.

### Integrated Drive Electronics (IDE) Controller

The onboard Enhanced Integrated Drive Electronics (EIDE) controller provided through the ALI chip set is an industry standard IDE interface for two disk drives or other peripherals. The IDE control signals are provided through the P2 connector for easy connection to a VME-6200 series disk drive module, or to an external disk drive setup using a VME-TB21.

### Small Computer Systems Interface (SCSI) Controller

The Symbios 53C860 Small Computer Systems Interface (SCSI) controller provides a Fast 20 SCSI (Ultra SCSI) compatible interface for hard disk drives or other SCSI peripheral devices. The SCSI control signals are provide through the P2 connector for painless connection to a VME-6200 series disk drive module, or to an external disk drive setup using a VME-TB21.



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

**Multi-Mode Video Adapter**
The V5B uses the Cirrus Logic CL-GD5436 video controller which is fully software configurable and can be set for compatibility with a wide variety of video displays. Operation with color monitors is usually achieved through emulation of IBM VGA or SVGA video. A standard 15-pin VGA connector is provided through the front panel for attaching VGA and SVGA capable monitors.

**On-Board Ethernet**
The V5B incorporates a high-performance DEChip 21143 Ethernet controller on-board. The V5B can be ordered with a Thinwire (10Base2) physical interface provided with a BNC connector through the front panel, or with Twisted-Pair (10/100BaseT) physical interface provided with an RJ45 connector through the front panel.

**PCI Bridge**
The V5B uses the DEC 21152 PCI to PCI bridge. This provides the user the ability to add PMC cards using SBS Embedded Computers' DPMCC or Industry Packs through SBS Embedded Computers' QIPC.

## VMEbus VME64 Compatibility Features

To meet the VME64 ANSI/VITA 1-1994 standard, the V5B uses the Tundra Universe II VMEbus Interface chip designed in consultation with Motorola. The VMEbus interface of the V5B is high-speed and very flexible, yet is easy for software to utilize. The following paragraphs discuss features of the Universe II VMEbus interface.

V5B revisions B0 and higher contain a Universe II chip, which is an updated version of the Universe II chip. You should have received a manual titled Universe II Register Definitions along with this manual. The Register Definitions manual is a copy of Appendix A of the Tundra Universe II manual, containing register information for programming. This manual may be obtained as a supplement to this technical manual from SBS Embedded Computers by emailing sales@ldg.com. The complete technical manual for the Tundra Universe II chip (including Appendix A) can be downloaded via the Internet from Tundra Semiconductor Corporation's website at http://www.tundra.com.

**Full Universe II VMEbus Master/Slave Operation**
The V5B was designed to provide both VMEbus Master and VMEbus Slave capability. As such, it can initiate data transfers over the VMEbus and receive data from other VMEbus Masters. A number of facilities have been provided to enhance VMEbus Master and Slave operation.

The VMEbus interface provides four VMEbus access paths. Each can be configured as either a VMEbus master or VMEbus slave path. A path configured as a slave allows access to the V5B System DRAM and other local resources from the VMEbus. The interface can be configured to allow all, part, or none of the local address space to be accessible, allowing local data to be configured as inaccessible from the VMEbus.



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

An access path configured as a master allows the V5B to perform data transfers on the VMEbus. A path can be configured to appear anywhere in local address space, and map anywhere on the VMEbus, providing for a very flexible interface. One path can be located in UMB space to allow VME access from programs operating in 80x86 Real Mode.

**Byte Swapping Capability**

The V5B features hardware implemented byte swapping. This allows for faster data transfers than possible using software implemented byte swapping. Supported modes of byte swapping are M32(no swapping), I32(Intel swapping), and I16(D16 byte swap). These modes can be specified for both master and slave transfers. CAUTION: I32 mode works only with D32 transfers. I16 works only with D16 transfers. Any other combinations with these two byte swap modes will result in invalid data.

**VMEbus System Controller Capability**

The V5B can be software configured as a VMEbus System Controller, independent of VMEbus Master or Slave operation. This is especially important if the V5B is going to be the only CPU card in the VMEbus system. Also, the V5B can automatically detect if it is in slot one and turn on the System Controller function.

**Interrupter and Interrupt Handler**

The Universe II chip provides a flexible scheme to map interrupts to either the PCI bus or the VMEbus. The Universe II Interrupter provides an 8-bit STATUS/ID to a VMEbus interrupt handler during the IACK cycle, and optionally generates an internal interrupt to signal that the interrupt vector has been provided. A VMEbus interrupt can trigger the Universe II to generate a normal VMEbus IACK cycle. When the IACK cycle is complete, the Universe II releases the VMEbus and the interrupt vector is received by the PCI resource servicing the interrupt output.



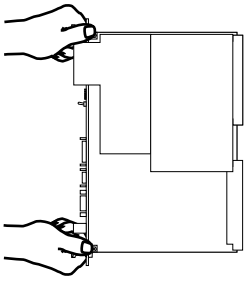
Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

CHAPTER 2

# Unpacking, Inspection, and Initial Operation

## Chapter Scope

This chapter covers the suggested inspection and preparation considerations and background information necessary prior to using the V5B. Unpacking, initial inspection, and first-time operation of the V5B are covered. Following the procedures given in the chapter is recommended, since they will verify proper operation after shipping and before the product is integrated into your VMEbus system.



Proper handling of the V5B is critical to ensure proper operation and long-term reliability. When unpacking the board, and whenever handling it thereafter, be sure to hold the V5B by the front panel or the card ejectors as shown in the drawing on this page. Do not hold the V5B by the circuit card edges, the heat sink, or the VMEbus connectors.

## Electrostatic Discharge Notice

The discharge of static electricity, known as electrostatic discharge or ESD, is a major cause of electronic component failure. The V5B has been packaged in a static-safe bag which protects the board from ESD while the board is in the bag. Before removing the V5B or any other electronic product from its static-safe bag, be prepared to handle it in a static-safe environment.


You should wear a properly-functioning antistatic strap and ensure you are fully grounded. Any surface upon which you place the unprotected V5B should be static-safe, usually facilitated by the use of antistatic mats. From the time the board is removed from the antistatic bag until it is in its card cage and functioning properly, extreme care should be taken to avoid “zapping” the board with ESD. You should be aware that you could “zap” the board without your knowing it; a small discharge, imperceptible to the eye and touch, is enough to damage electronic components. Extra caution should be taken in cold and dry weather when static easily builds up.

## Unpacking and Handling

The V5B has been carefully packaged to ensure adequate protection from the rigors of shipping. However, the possibility exists that shipping damage can occur. Close inspection of the shipping carton should reveal some information about how the package was handled by the shipping service. If evidence of damage or excessively rough handling is found, you should notify the shipping service and SBS Embedded Computers as soon as possible.



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

Only after ensuring that both you and the surrounding area are protected from ESD, carefully remove the V5B from the shipping carton by grasping it by the front panel and the VME connectors. Place the board, in its anti-static bag, flat down on a suitable surface. You may then remove the board from the anti-static bag by tearing the ESD warning labels. Again place the board flat down on the work surface.

## Initial Inspection

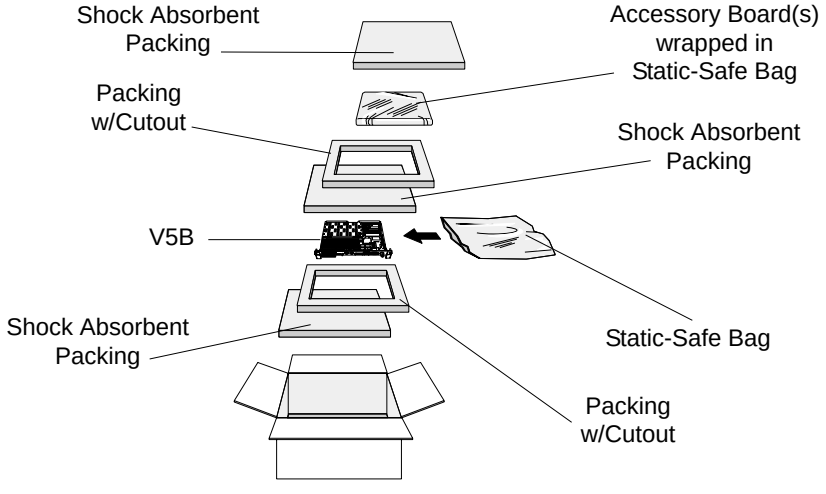
Once the V5B has been unpacked, it is suggested you inspect it for visible damage which could have occurred during shipping or unpacking. If there is visible damage, usually in the form of bent component leads or loose socketed components, contact SBS Embedded Computers for additional instructions. Depending on the severity of the damage, it may need to be sent back to the factory to be repaired. DO NOT apply power to the board if it has visible damage. Doing so may cause further, possibly irreparable damage, as well as introduce a fire or shock hazard. Since the V5B incorporates a number of socketed components, including the CPU, memory, Ethernet adapter, these should be inspected to make sure that they are seated fully in their sockets.

## Included Items

Table 2-1 lists accessories which are included with each V5B. You may want to make sure that you have the items shown in the event that you need them during installation or configuration.

| Qty. | Item | Purpose |
|---|---|---|
| 1 | Cable Adapter | PC Keyboard Adapter |

Table 2-1 Included Items




*Retain all packing materials in case of future need for storage, relocation or the possibility of return for repairs.*



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

## Minimum System Requirements

As shipped, the V5B has been thoroughly tested and burned-in, and is nearly ready for use in your VMEbus system. In order to verify V5B operation for the first time, it is suggested that you only configure a minimal system. It is not necessary to have disk drives or other accessories connected in order to perform the V5B confidence tests. The following items are all that are required:

### VMEbus Backplane and Power Supply

You will need a standard VMEbus P1 or combination P1/P2 backplane wired into a regulated power supply capable of providing stable low noise +5 volt, +12 volt, and -12 volt sources. Make sure that the supply is capable of meeting the total power requirements of the V5B. Please refer to chapter 12 for the V5B Specifications. Initially, you may plug the V5B into any available 6U slot of your VMEbus backplane. Later you will need to select the proper slot for your particular application. Please make sure that you do not have the power supply turned ON when the V5B is plugged into your backplane.

***As a reminder, you must never install or remove any VMEbus card while power is applied unless it is certified for hot-insertion (the V5B is not).***

### Keyboard

You should have an IBM PC/AT or compatible keyboard for initial system operation. Depending on your application, this keyboard may be a standard full-travel keyboard, or one which utilizes membrane switches for harsh environments. Note that some keyboards have a switch which selects either PC and PC/XT compatibility or PC/AT compatibility. If your keyboard is equipped with such a switch, make sure it is set for PC/AT compatibility. The miniature DIN keyboard connector is located on the front panel and is directly compatible with IBM PS2-type keyboards. If your keyboard has the normal IBM PC-type keyboard connector you will need to use the supplied cable adapter. Plug your keyboard into the keyed front panel connector labeled ‘KEYBOARD’ before power is applied. Plugging and unplugging the keyboard while power is applied is not recommended and can cause a software failure or damage to the keyboard or the on-board circuitry.

### Video Monitor

Any VGA-compatible video monitor can be used initially for display output. The 15-pin front panel connector is fully compatible with the VGA standard connector and pinout. Plug your video monitor into the connector labeled ‘VIDEO’.

After making sure that you have installed the V5B properly into your VMEbus backplane you may apply power to your video monitor and then to the VMEbus supply. When the board is completely reset, the processor should begin executing initial BIOS-resident routines.



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

## Initial Power-On Operation

After a few seconds, the V5B System BIOS banner will be displayed similar to that shown in Figure 2-1.

If you have seen all the messages to this point, you can be confident that the board is running properly and is ready to be installed and setup for your application. You may now shut off power to the VMEbus chassis and begin board setup and configuration as described in Chapter 3.

```
Award Modular BIOS
Copyright (C)  1984-98, Award Software, Inc.

SBS Embedded Computers - V5B Bios V1.04

AMD K6™ - 2/333 CPU found
Memory Test: 65536k OK
```

Figure 2-1 System BIOS Banner

If the board did not perform as described above, some damage may have occurred in shipping or the board is not installed or setup properly. Contact SBS technical support as described in Chapter 12 for further instructions.



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

# CHAPTER 3 Installation and Interfacing

## Chapter Scope

This chapter covers installing the V5B in a VMEbus system and interfacing the V5B to external devices and peripherals.

## Installing the V5B

The V5B is designed to conform to the VMEbus physical specifications for a 6U×4HP dual eurocard (dual-height, single-width). It can be plugged directly into any standard card cage designed for these types of cards. Installation of the V5B is relatively simple as described below.

## Installation Requirements

The following paragraphs describe requirements which your VMEbus system must meet in order to install and successfully integrate the V5B. If you are uncertain about any requirements, be sure to contact SBS Embedded Computers Technical Support before installing your V5B.

### VMEbus Power Requirements

Power is supplied to the V5B via the VMEbus P1 connector as shown in Table 3-1.

| Signal | Location on VMEbus P1 |
|---|---|
| +5 Volts | Row a, pin 32 |
| | Row b; pin 32 |
| | Row c; pin 32 |
| Ground | Row a; pins 9, 11, 15,17,19 |
| | Row b; pins 20,23 |
| | Row c; pin 9 |
| +12 Volts | Row c; pin 31 |
| -12 Volts | Row a; pin 31 |

Table 3-1 VMEbus P1 Power Pins



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

If you are using the V5B in a full 32-bit address system, additional power and ground pins are connected according to VMEbus specification for P2 row *b* as shown in Table 3-2.

| Signal | Location on VMEbus P2 |
|---|---|
| +5 Volts | Row b; pins 1,13,32 |
| GND | Row b; pins 2,12,22,31 |

Table 3-2 VMEbus P2 Power Pins

The actual electrical requirements of the V5B are defined in the Specifications section of chapter 12.

**WARNING:**

*The P2 connector of the V5B has the a and c rows defined by SBS as the VMEbus specification permits. However, other boards on the VMEbus may also define these rows. Damage can result to the V5B or other boards if the P2 a and c row signals are bussed between boards which are not compatible.*

### VMEbus Backplane Requirements

Since the V5B Embedded PC is capable of generating 16- or 32-bit data and 16-, 24- or 32-bit addresses it can be used in VMEbus systems which have either a single P1 backplane or a dual P1/P2 backplane.

### VMEbus P1 Connector

The V5B has been designed to directly interface with any VMEbus P1 backplane conforming to the Rev. C.1 specification. Table 3-3 summarizes the VMEbus P1 signals used.



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

| Pin | Row a Signal | Row b Signal | Row c Signal |
|---|---|---|---|
| 1 | D0, Data bit 0 | BBSY- | D8, Data bit 8 |
| 2 | D1, Data bit 1 | BCLR | D9, Data bit 9 |
| 3 | D2, Data bit 2 | ACFAIL- | D10, Data bit 10 |
| 4 | D3, Data bit 3 | BG0IN- | D11, Data bit 11 |
| 5 | D4, Data bit 4 | BG0OUT- | D12, Data bit 12 |
| 6 | D5, Data bit 5 | BG1IN- | D13, Data bit 13 |
| 7 | D6, Data bit 6 | BG1OUT- | D14, Data bit 14 |
| 8 | D7, Data bit 7 | BG2IN- | D15, Data bit 15 |
| 9 | GND | BG2OUT- | GND |
| 10 | SYSCLK | BG3IN- | SYSFAIL- |
| 11 | GND | BG3OUT- | BERR- |
| 12 | DS1- | BR0- | SYSRESET- |
| 13 | DS0- | BR1- | LWORD- |
| 14 | WRITE- | BR2- | AM5 |
| 15 | GND | BR3- | A23 |
| 16 | DTACK- | AM0 | A22 |
| 17 | GND | AM1 | A21 |
| 18 | AS- | AM2 | A20 |
| 19 | GND | AM3 | A19 |
| 20 | IACK- | GND | A18 |
| 21 | IACKIN- | Not used | A17 |
| 22 | IACKOUT- | Not used | A16 |
| 23 | AM4 | GND | A15 |
| 24 | A07 | IRQ7- | A14 |
| 25 | A06 | IRQ6- | A13 |
| 26 | A05 | IRQ5- | A12 |
| 27 | A04 | IRQ4- | A11 |
| 28 | A03 | IRQ3- | A10 |
| 29 | A02 | IRQ2- | A09 |
| 30 | A01 | IRQ1- | A08 |
| 31 | -12V | +5VSTDBY | +12V |
| 32 | +5V | +5V | +5V |

Table 3-3 V5B P1 Signals

### VMEbus P1 Daisy Chain Signals

The VMEbus specification defines two signal paths which are connected in daisy chain fashion, meaning that they are intercepted at each card position before being passed on the the adjacent card. These signals are associated with bus acquisition and interrupt acknowledgement. Before plugging the V5B into the P1 backplane, you should make sure that the backplane slot is configured appropriately for these signals.



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

### Bus Grant Daisy Chain

The VMEbus has four bus grant daisy chain signal pairs: BG0IN-/BG0OUT-, BG1IN-/BG1OUT-, BG2IN-/BG2OUT-, BG3IN-/BG3OUT-. When configuring a VMEbus P1 backplane, it is necessary to jumper all bus grant daisy chain signals on the backplane so that the bus grant outputs are connected to the bus grant inputs at all unused slots. At the slot in which a V5B is installed none of the bus grant signals should be jumpered. Most VMEbus P1 backplanes provide an easy method of jumpering the bus grant signals at each card slot position.

### Interrupt Acknowledge Daisy Chain

The VMEbus has a single interrupt acknowledge daisy chain signal pair; IACKIN-/IACKOUT-. As the V5B has an on-board VMEbus interrupter, it utilizes these signals for determining that a VMEbus acknowledge cycle in progress is intended for it. Therefore, it is necessary to make sure that the IACKIN- is not connected to the IACKOUT- at the slot where the V5B is to be installed. During an interrupt acknowledge cycle, the V5B will intercept the IACKIN- signal and either respond if it has a pending interrupt at the proper level, or it will pass the signal on to the next card via the IACKOUT- signal line. The interrupt controller in non-Slot 1 applications requires the backplane to have a “wrap-around” path for IACK- to Slot 1. Most VMEbus P1 backplanes provide an easy method of setting the interrupt acknowledge daisy chain at each slot.

### The Standard P2 Connector

Row *b* of the V5B P2 connector conforms to the VMEbus specification for A32/D32 operation. SBS Embedded Computers has defined the *a* and *c* rows of the P2 connector with the signals for SCSI, IDE hard disk, and ST412 floppy disk interfaces. This is perfectly acceptable and in accordance with VMEbus specifications. SBS Embedded Computers has a TB-21 Translation Board which can adapt the row *a* and *c* signals of the P2 to standard SCSI, IDE hard disk, and ST412 floppy disk connectors. Also, SBS Embedded Computers has several disk drive modules which can directly connect to the V5B P2 via a P2-to-P2 cable. This system avoids cables in the card space of a card cage, and provides flexibility as to the placement of drives and other peripherals. The pinout of the V5B P2 connector is shown inTable 3-4.

**WARNING:** *The a and c row definitions of P2 on a V5B are not compatible with the a and c row definitions of P2 on a VME-1486. As such, the transition boards and disk drive modules available for the VME-1486 cannot be used with the V5B. Connecting a VME-1486 P2 transition board or drive module to a V5B will void your warranty and may cause possibly irreparable damage to the V5B, the peripheral, or both.*



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

| Pin | Row a Signal | Row b Signal | Row c Signal |
|---|---|---|---|
| 1 | DB0 | +5V | HDRESET- |
| 2 | DB1 | GND | HD0 |
| 3 | DB2 | Reserved | HD1 |
| 4 | DB3 | A24 | HD2 |
| 5 | DB4 | A25 | HD3 |
| 6 | DB5 | A26 | HD4 |
| 7 | DB6 | A27 | HD5 |
| 8 | DB7 | A28 | HD6 |
| 9 | PARITY- | A29 | HD7 |
| 10 | ATN- | A30 | HD8 |
| 11 | BSY- | A31 | HD9 |
| 12 | ACK- | GND | HD10 |
| 13 | RST- | +5V | HD11 |
| 14 | MSG- | D16 | HD12 |
| 15 | SEL- | D17 | HD13 |
| 16 | C/D- | D18 | HD14 |
| 17 | REQ- | D19 | HD15 |
| 18 | I/O- | D20 | PDIAG- |
| 19 | HDMARQ | D21 | HDIOW- |
| 20 | FRWC- | D22 | HDIOR- |
| 21 | FINDEX- | D23 | IOCHRDY |
| 22 | FMT0- | GND | HDMACK- |
| 23 | FDS2- | D24 | HDIRQ |
| 24 | FDS1- | D25 | IOCS16- |
| 25 | FMT1- | D26 | HDA0 |
| 26 | FDIRC- | D27 | HDA1 |
| 27 | FSTEP- | D28 | HDA2 |
| 28 | FWD- | D29 | HCS0- |
| 29 | FWE- | D30 | HCS1- |
| 30 | FTK0- | D31 | SLVACT- |
| 31 | FWP- | GND | FHS- |
| 32 | FRD- | +5V | FDSKCHG- |

SCSI
IDE
IDE
Floppy
Floppy

Table 3-4 V5B P2 Signals

### Cooling Requirements

The V5B integrates a large number of high-performance components in the small space of a single-slot VME card. These components generate a lot of heat. To prevent damage to the board and to ensure reliable operation, the V5B MUST be cooled with forced air. A 100LFM fan is recommended.

## Interfacing the V5B to External Devices

Several connectors are provided on-board the V5B in order to establish an electrical interconnection to external devices. The normal PC keyboard, serial, parallel and video connections are provided through the front panel where they may be easily accessed while the card is installed in a card cage. In addition, the COM-2 port has software selectable RS-422 and RS-485 signals, Ethernet is provided through the front panel, and the standard PC/AT bus signals are provided via the two expansion interface connectors at the bottom edge of the board.



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

## Keyboard Interface

The keyboard interface consists of four electrical connections provided on a 6-pin circular mini-DIN connector (PS/2-style). Two connections are for communication between the keyboard and the keyboard interface port. Both of these signals are CMOS compatible. Two connections provide power for the keyboard's operation. The keyboard connector with pin assignments as viewed from the front of the V5B are shown in Figure 3-1. Keyboard signal definitions are given in Table 3-5.

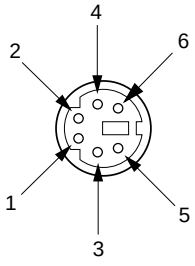



Figure 3-1 Keyboard Interface

| Pin | Signal | Function |
|---|---|---|
| 1 | KBDAT | Bidirectional serial data to/from keyboard |
| 2 | NC | Not used, no connection made |
| 3 | GND | Digital Ground (0 volt) reference to keyboard |
| 4 | VCC | +5 volt power to the keyboard |
| 5 | KBCLK | Bidirectional data clock to/from keyboard |
| 6 | NC | Not used, no connection made |

Table 3-5 Keyboard Interface Signals

This connector accepts any manufacturer's keyboard which is designed with a standard PC/AT interface; however, a PC/AT to PS/2 adapter is required. Such an adapter is supplied with the board. Note that many keyboard manufacturers provide keyboards which will work with either the IBM PC or PC/AT. Most of these have a switch which the user must set to match the type of computer that its used with. For the V5B, make sure that the switch is set for PC/AT operation, not for PC or PC/XT operation.

When plugging a keyboard into the V5B's keyboard connector, it is recommended that the system power be switched OFF. Plugging a keyboard into a powered-up system may cause improper initialization of the keyboard. If there is a static charge on the keyboard, it may cause damage to the V5B's keyboard controller circuitry or to the keyboard circuitry itself.



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

## PS/2 Mouse Interface

The PS/2 mouse interface is a 6-pin circular mini-DIN connector almost identical to the keyboard interface. Two connections are for communication between the mouse and the interface port. Both of these signals are CMOS compatible. Two connections provide power for operation of the mouse. The PS/2 mouse connector has the same pin number assignments as the keyboard shown in Figure 3-1. Mouse signal definitions are given in Table 3-6.

| Pin | Signal | Function |
|---|---|---|
| 1 | MSDATA | Bidirectional serial data to/from mouse |
| 2 | NC | Not used, no connection made |
| 3 | GND | Digital Ground (0 volt) reference to mouse |
| 4 | VCC | +5 volt power to the mouse |
| 5 | MSCLK | Bidirectional data clock to/from mouse |
| 6 | NC | Not used, no connection made |

Table 3-6 PS/2 Mouse Interface Signals

## Serial Ports

The V5B is equipped with two serial ports which are compatible with the COM-1 and COM-2 ports on a standard IBM PC. Its signals are defined for industry standard interfacing using RS-232 compatible signal levels. Because of panel space it is pinned out on a 9-pin male micro-miniature D-type connector according to the standard IBM PC pinout. COM-1 and COM-2 are stacked and are located just above the video connector. The connector pins are identified in Figure 3-2.

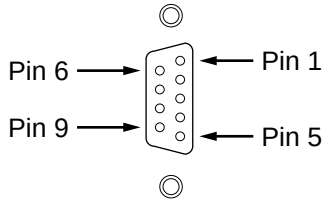



Figure 3-2 Serial Communications (COM) Port

Table 3-7 defines the signals present on the pins of the COM-1 and COM-2 port connector, their intended function and their direction.

An adapter cable to convert from the micro-miniature D to a standard 9-pin D is available from SBS Embedded Computers. The part number is CA9S24.



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

| Pin | Signal | Direction | Function |
|---|---|---|---|
| 1 | DCD | To V5B | Data Carrier Detect, from modem |
| 2 | RXD | To V5B | Received Data, from modem |
| 3 | TXD | From V5B | Transmitted Data, to modem |
| 4 | DTR | From V5B | Data Terminal Ready, to modem |
| 5 | SIGND | — | Serial Interface Ground reference |
| 6 | DSR | From V5B | Data Set Ready, from modem |
| 7 | RTS | From V5B | Request To Send, to modem |
| 8 | CTS | To V5B | Clear To Send, from modem |
| 9 | RI | To V5B | Ring Indicator, from modem |

Table 3-7 COM-1 and COM-2 (RS-232 Mode) Pin Assignments

## Extended COM-2 Functionality

The COM-2 serial port of the V5B provides functionality beyond that of a standard PC/AT. Under software control, COM-2 can be configured for RS-422 or RS-485 operation in place of standard RS-232. RS-422 operation on the V5B is four-wire full duplex. RS-485 operation on the V5B is two-wire half duplex. Table 3-8 shows the COM-2 signals for RS-422 operation. Table 3-9 shows the COM-2 signals for RS-485 operation.

| Pin | Signal | Direction | Function |
|---|---|---|---|
| 1 | NC | — | No connection |
| 2 | RXD- | To V5B | Receive data, negative |
| 3 | TXD- | From V5B | Transmit data, negative |
| 4 | TXD+ | From V5B | Transmit data, positive |
| 5 | SIGND | — | Serial Interface Ground reference |
| 6 | RXD+ | To V5B | Receive data, positive |
| 7 | NC | — | No connection |
| 8 | NC | — | No connection |
| 9 | NC | — | No connection |

Table 3-8 COM-2 RS-422 Mode Pin Assignments

In RS-422 or RS-485 mode it may be necessary to provide termination at the V5B if it is at either end of a multidrop link. For RS-485 this is be done automatically by the V5B. For RS-422, terminating resistors must be installed in the external connector shield. The terminating resistors should result in a low impedance of 100 ohms or less.

Note that the DSR and CTS signals are always true which allows DOS drivers and communications software to work properly. Also, the RTS line, when true, enables the RS-422 transmitter when V5B RS-422/485 operation is enabled.



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

| Pin | Signal | Direction | Function |
|---|---|---|---|
| 1 | NC- | — | No connection |
| 2 | TX/RX- | From/To V5B | Transmit/receive data, negative |
| 3 | NC | — | No connection |
| 4 | NC | — | No connection |
| 5 | SIGND- | — | Serial Interface Ground reference |
| 6 | TX/RX+ | From/To V5B | Transmit/receive data, positive |
| 7 | NC | — | No connection |
| 8 | NC | — | No connection |
| 9 | NC | — | No connection |

Table 3-9 COM-2 RS-485 Half-Duplex Mode Pin Assignments

## Parallel Port

The parallel port on the V5B is accessible through a 25 pin male micro-miniature D-type connector, which appears just below the keyboard port on the V5B front panel. In its normal operating mode, the port functions exactly as does a standard parallel printer port on a normal IBM PC. In this mode of operation, it will interface with printers utilizing the industry standard Centronics parallel interface. Figure 3-3 is the connector pinout and signals are defined in Table 3-10.

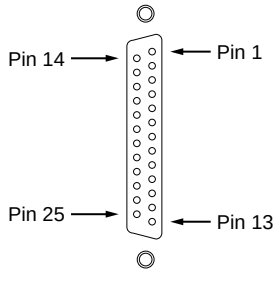



Figure 3-3 Parallel Port

| Pin | Signal | Direction | Function |
|---|---|---|---|
| 1 | STROBE- | Output | Data strobe |
| 2 | PD0 | Output | Data bit 0 |
| 3 | PD1 | Output | Data bit 1 |
| 4 | PD2 | Output | Data bit 2 |
| 5 | PD3 | Output | Data bit 3 |
| 6 | PD4 | Output | Data bit 4 |
| 7 | PD5 | Output | Data bit 5 |
| 8 | PD6 | Output | Data bit 6 |
| 9 | PD7 | Output | Data bit 7 |
| 10 | ACK- | Input | Data acknowledge |
| 11 | BUSY | Input | Printer busy |
| 12 | PE | Input | Printer enabled |
| 13 | SLCT | Input | Printer selected |
| 14 | AUTOFD- | Output | Automatic feed |
| 15 | ERROR- | Input | Printer error indicator |
| 16 | INIT- | Output | Initialize |
| 17 | SLCTIN- | Output | Printer select |
| 18–25 | GND | — | Signal Ground |

Table 3-10 Parallel Port Pin Assignments



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

A 24-inch adapter cable to convert from the micro-miniature D to a standard 25-pin D is available from SBS Embedded Computers. The part number is CA25P24. Note that the parallel printer interface is TTL level compatible, therefore limiting operation over a distance of eight feet or less.

Operation of the printer port is accomplished with routines built into the ROM BIOS supplied with the V5B. Therefore, standard MS-DOS printer functions will work with a printer attached as the “PRN” device.

Under software control, the parallel port supports the optional PS/2 type bidirectional parallel port (SPP), the Enhanced Parallel Port (EPP), and the Extended Capabilities Port (ECP) modes. Refer to the SMC FDC37C665GT manual for configuration information.

## Video Interface

The video controller provided on-board the V5B Embedded PC is capable of supporting a variety of VGA-compatible video devices, such as VGA and SVGA CRTs. The standard 15-pin VGA connector is provided through the front panel.

## Ethernet Interface

The V5B contains a high performance DEChip 21143 Ethernet controller, and comes standard with a Thinwire (10Base2) interface with a BNC connector through the front panel. It can be ordered with a Twisted Pair (10/100BaseT) interface with an RJ45 connector through the front panel. Table 3-11 shows the pin assignments for the RJ45, and Figure 3-4 indicates the BNC connector.

| Pin | Signal |
|---|---|
| 1 | Transmit Data Positive |
| 2 | Transmit Data negative |
| 3 | Receive Data Positive |
| 4 | No connection |
| 5 | No connection |
| 6 | Receive Data Negative |
| 7 | No connection |
| 8 | No connection |

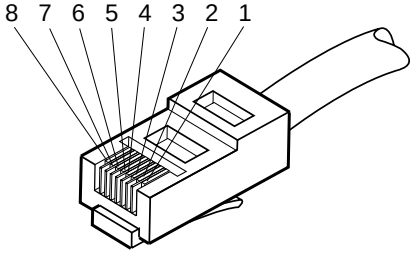



Table 3-11 Ethernet, 10/100BaseT, RJ45 Pin Assignments

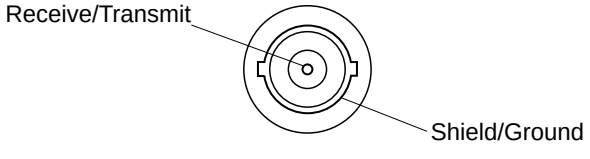



Figure 3-4 Ethernet, 10Base2 Connector



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

# CHAPTER 4 Booting and Operation

## Chapter Scope

This chapter describes the Award Software BIOS Setup program. The Setup program allows you to modify basic system configuration settings. The settings are stored in a dedicated battery-backed memory, called CMOS RAM, that retains the information when the power is turned off.

The Award Software BIOS is immediately activated when you first turn on the computer. The BIOS reads system configuration information in CMOS RAM and begins the process of checking out the system and configuring it through the Power-On Self Test (POST). The BIOS then seeks an operating system on one of the data storage devices (hard drive, floppy drive, etc.). The BIOS launches the operating system and hands control of system operations to it. You can enter the Setup program by pressing <DEL> immediately after switching the system on.

## In Case of Problems

If you should discover that your computer is no longer able to boot after making and saving system changes with Setup, the BIOS supports an override to the CMOS settings that resets your system to its default configuration. You can invoke this override by immediately pressing <Del> when you restart your computer. You can restart by either using the ON/OFF switch, the RESET button, or by pressing Ctrl-Alt-Del.

It is recommended that you only alter settings that you thoroughly understand.



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

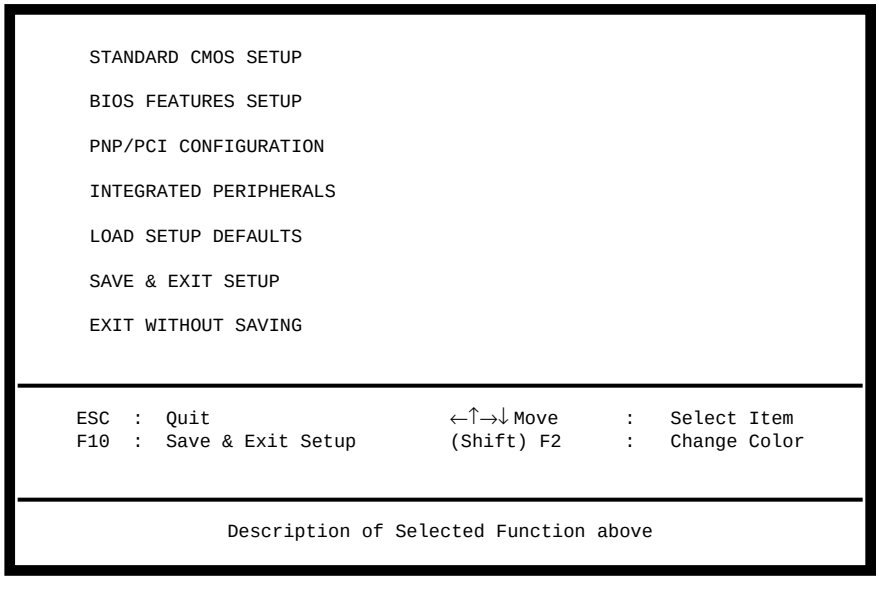
## Setup Key Commands

The following keys may be used to operate the Setup program:

| | |
|---|---|
| **Arrow Keys** | The arrow keys are used to move around the Setup screen. The up and down arrows move you through the previous and the next options respectively. The right arrow moves the cursor into the right side of the screen. The left arrow moves the cursor into the left side of the screen. |
| **ESC** | In the Main Menu, pressing the escape key exits the Setup program without saving changes. In other screens, it will exit current page and return to main menu. |
| **PgUp** or **+** | Increase the numeric value, or make changes. |
| **PgDn** or **-** | Decrease the numeric value, or make changes. |
| **F1** | General help only for Status page and Option page Setup Menu. |
| **F2** | Change color from total 16 colors. F2 to select Shift-F2 color forward, Shift-F2 color backward. |
| **F5** | Restores the previous CMOS value from CMOS, only for Option Page Setup Menu.. |
| **F6** | Loads Default BIOS values. (Option Page Setup Menu) |
| **F10** | Saves all CMOS changes. (Main Menu only) |

## Main Setup Menu

```
ROM PCI/ISA BIOS (2A5KFU80)
     CMOS SETUP UTILITY
   AWARD SOFTWARE, INC.
```



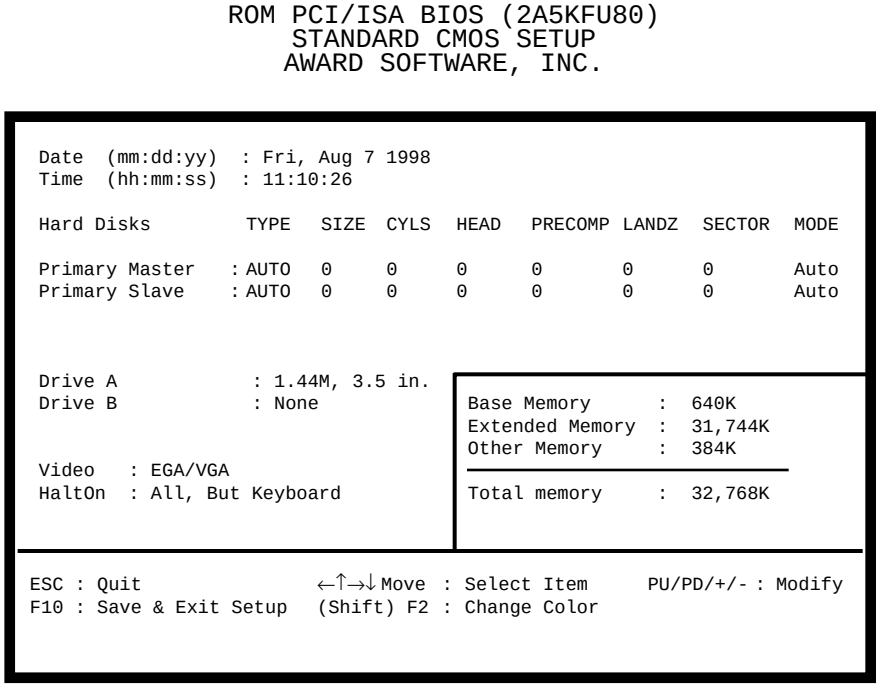



Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

| | |
|---|---|
| **Standard CMOS** | Options in the original PC AT compatible BIOS. |
| **BIOS Features** | Award Software enhanced BIOS features. |
| **PNP/PCI Configuration** | Local bus configuration options. |
| **Integrated Peripherals** | Settings for HDD, controllers, serial ports and other peripheral devices. |
| **Load Setup Defaults** | Factory settings for the most common options. |
| **Save and Exit Setup** | Save settings in non-volatile CMOS RAM and exit Setup. |
| **Exit Without Save** | Abandon all changes and exit Setup. |

The following screens detail additional options available with the respective selections from the Main Menu screen.

## Standard CMOS Setup






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

**Date** This field is for information only, and can't be altered.

**Time** Time format is based on the 24-hour military clock. For example, 1 pm is 13:00. Pres right or left arrow to to move to desired fileds. Press PgUp or PgDn to increment the setting, or type in the desired value.

**Hard Disks** The BIOS supports up to two IDE drives. This section does not show information about other IDE drives, such as a CD ROM drive, or other hard drive types, such as SCSI drives. It is recommended that you select type AUTO for all drives. The BIOS can automatically detect the specifications and optimal mode for almost all IDE hard drives. When you select type AUTO for a hard drive, the BIOS detects its specifications during POST, each time the system boots. If you do not want to select type AUTO here, you may do one of the following:

1.) Match the specifications of your IDE hard drive(s) with the preprogrammed values for types 1 thru 45.
2.) Select USER and enter values directly into each drive parameter field.

**Drive A, Drive B** Select the correct specifications for the diskette drive(s) installed in your system.

**Video** Select the type of primary video subsytem in your computer. The BIOS usually detects the correct video type automatically.

**Halt ON** During the power-on self-test (POST), the computer stops if the BIOS detects a hardware error. You can tell the BIOS to ignore certain errors during POST and continue the boot process. These error selections are:

**No Errors** POST does not stop for any errors.

**All Errors** If the BIOS detects any non-fatal errors, POST stops and prompts you to take corrective action.

*All But Keyboard* POST does not stop for a keyboard error, but stops for all other errors.

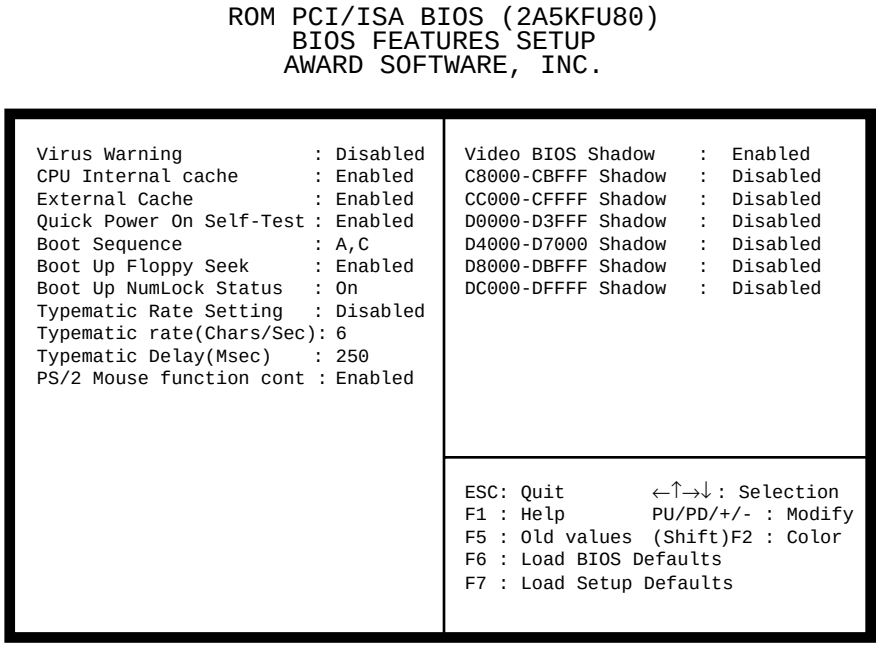
*All But Diskette* POST does not stop for diskette drive errors, but stops for all other errors.

*All But Disk/Key* POST does not stop for a keyboard or disk error, but stops for all other errors.

**Memory** You cannot change any values in the Memory fields. They are there for information only.



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




**Virus Warning**

When enabled, you receive a warning message if a program (specifically, a virus) attempts to write to the boot sector or the partition table of the hard disk drive. You should then run an anti-virus program. This only protects the boot sector, not the entire hard drive.
***NOTE:*** *Many disk diagnostic programs that access the boot sector table can trigger the virus warning message. If you plan to run such a program, it is recommended that you disable the virus warning first.*

**CPU Internal Cache /External Cache**

Cache memory is additional memory that is much faster than conventional DRAM (system memory). When the CPU requests data, the system transfers the requested data from the main DRAM into cache memory, for even faster access by the CPU.

**Quick Power On Self-Test**

Select enabled to to reduce the amount of time required to run the power-on self-test (POST). A quick POST skips certain steps. It is recommended that you normally disable Quick POST. It is better to find a problem during POST than lose data during your work.

**Boot Sequence**

The original IBM PCs loaded the DOS fromdrive A (floppy disk), so IBM PC-compatible systems are designed to search for an operating system first on drive A, and then on drive C (hard disk). Most modern computers load the operating system from the hard drive, and may even load it from a CD ROM drive. The options are: A,C; C,A; C,CDROM,A; CDROM,A,A; C only, LS120,C



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

| | |
|---|---|
| **Boot Up Floppy Seek** | When enabled, the BIOS tests (seeks) floppy drives to determine whether they have 40 or 80 tracks. Only 360k drives have 40 tracks. 720k, 1.2M, and 1.44M drives all have 80 tracks. Because very few modern PCs have 40-track drives, it is recommended that you set this field to DISABLED to save time. |
| **Boot Up NumLock Status** | Toggle between ON and OFF to control the state of the NumLock key when the system boots. When toggled On, the numeric keypad generates numbers instead of controlling cursor operations. |
| **Typematic Rate Setting** | When DISABLED, Typematic Rate and Typematic Delay are irrelevant. Keyboard repeats at a rate determined by the keyboard controller in your system. When this is ENABLED, you can select Typmatic Rate and Delay. |
| **Typematic Rate (Chars/Sec)** | When the Typematic Rate setting is ENABLED, you can select a typematic rate (the rate at which a character repeats when you hold down a key) of 6, 8, 10, 12, 15, 20, 24,or 30 characters per second. |
| **Typematic Delay (msec)** | When the Typematic Rate setting is ENABLED, you can select a typematic delay (the delay before key strokes begin to repeat)of 250, 500, 750, or 1000 milliseconds. |
| **PS/2 Mouse Function Control** | Enables or Disables PS/2 Mouse function control. |

**Shadow**

Software residing in a Read-Only Memory (ROM) is called Firmware. The Award Software BIOS permits *Shadowing* of firmware such as system BIOS, video BIOS and similar operating instructions that come with some expansion peripherals such as a SCSI adapter. Shadowing copies from ROM into system RAM, where the CPU can read it through the 64-bit DRAM bus. Firmware not shadowed must be read by the system through the 8-bit X-bus. Shadowing improves the performance of the system BIOS and similar ROM firmware for expansion peripherals.

ENABLE shadowing into each section of memory separately.

Video BIOS shadows into memory area C0000-C7FFF. The remaining areas shown on the BIOS features Setup screen may be occupied by other expansion card firmware. If an expansion peripheral in your system contains ROM-based firmware, you need to know the address range the ROM occupies to shadow it into the correct of RAM.



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

```
ROM PCI/ISA BIOS (2A5KFU80)
PNP/PCI CONFIGURATION
AWARD SOFTWARE, INC.
```

```
Resources Controlled by    : Auto
Reset Configuration Date   : Disabled

ESC: Quit       ←↑→↓ :Select
F1 : Help       PU/PD/+/- : Modify
F5 : Old values (Shift)F2 : Color
F6 : Load BIOS Defaults
F7 : Load Setup Defaults
```

**Resources Controlled By** The Award plug and play BIOS can automatically configure plug-and-play devices. AUTO selection displays the screen as above. MANUAL introduces additional options as indicated below. These additional options are automatically handled in AUTO selection.

```
Resources Controlled by :Manual
Reset Configuration Data:Disabled
IRQ-3 Assigned To       :Disabled
IRQ-4 Assigned To       :PCI/ISA PnP
IRQ-5 Assigned To       :PCI/ISA PnP
IRQ-6 Assigned To       :PCI/ISA PnP
IRQ-7 Assigned To       :PCI/ISA PnP
IRQ-8 Assigned To       :PCI/ISA PnP
IRQ-9 Assigned To       :PCI/ISA PnP
IRQ-10 Assigned To      :PCI/ISA PnP
IRQ-11 Assigned To      :PCI/ISA PnP
IRQ-12 Assigned To      :PCI/ISA PnP
IRQ-13 Assigned To      :PCI/ISA PnP
IRQ-14 Assigned To      :PCI/ISA PnP
IRQ-15 Assigned To      :PCI/ISA PnP
DMA-0 Assigned To       :PCI/ISA PnP
DMA-1 Assigned To       :PCI/ISA PnP
DMA-2 Assigned To       :PCI/ISA PnP
DMA-3 Assigned To       :PCI/ISA PnP
DMA-4 Assigned To       :PCI/ISA PnP
DMA-5 Assigned To       :PCI/ISA PnP
DMA-6 Assigned To       :PCI/ISA PnP
DMA-7 Assigned To       :PCI/ISA PnP

ESC: Quit       ←↑→↓ :Select
F1 : Help       PU/PD/+/- : Modify
F5 : Old values (Shift)F2 : Color
F6 : Load BIOS Defaults
F7 : Load Setup Defaults
```

**Reset Configuration Data** This is normally left DISABLED. Select ENABLED to reset Extended System Configuration Data (ESCD) when you exit Setup if you have installed a new add-on and the system re-configuration has caused such a serious conflict that the operating system cannot boot.



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

**IRQ x Assigned To** When resources are controlled manually, assign each system interrupt as one of the following types, depending on the type of device using the interrupt:

LEGACY ISA:

Devices compliant with the original PC AT bus specification, requiring a specific interrupt (such as IRQ4 for serial port 1).

PCI/ISA PnP:

Devices compliant with the Plug and Play standard, whether designed for PCI or ISA bus architecture.

**DMA x Assigned To** When resources are controlled manually, assign each system DMA channel as one of the following types:

LEGACY ISA:

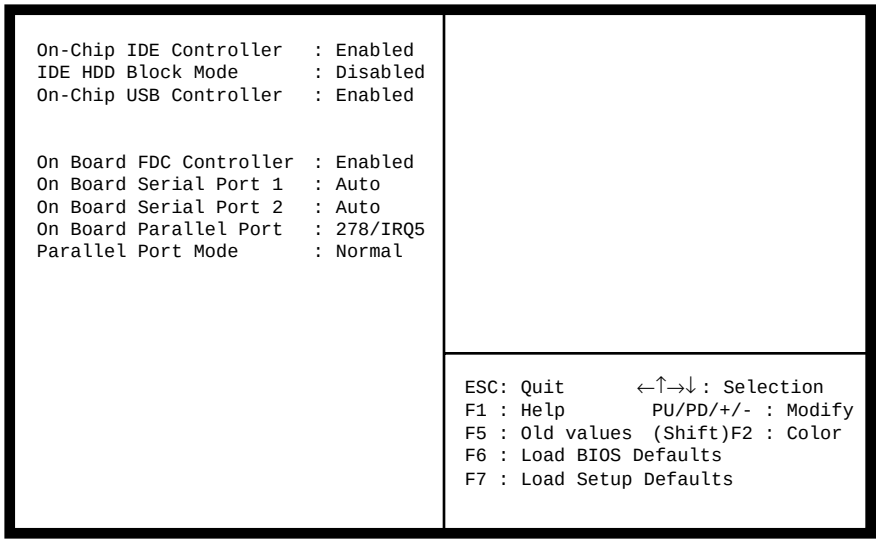
Devices compliant with the original PC AT bus specification, requiring a specific DMA channel.

PCI/ISA PnP:

Devices compliant with the Plug and Play standard, whether designed for PCI or ISA bus architecture.

## Integrated Peripherals

```
ROM PCI/ISA BIOS (2A5KFU80)
INTEGRATED PERIPHERALS
AWARD SOFTWARE, INC.
```




**On-Chip IDE Controller** The integrated peripheral controller contains an IDE interface with support for two IDE channels. Select ENABLED to activate the IDE interface.



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

| | |
|---|---|
| **IDE HDD Block Mode** | Block mode is also called block transfer, multiple commands, or multiple sector read/write. If your IDE drive supports block mode (most new drives do), select ENABLED for automatic detection of the optimal number of block read/writes per sector the drive can support. |
| **Onboard FDC Controller** | Select ENABLED if your system has a floppy disk controller (FDC) installed on the system board and you wish to use it. If you install an add-in FDC or the system has no floppy drive, select DISABLED in this field. |
| **Onboard Serial Ports (1/2, A/B)** | Select a logical COM port name and matching address for the first and second serial ports. Select an address and corresponding interrupt for the first and second serial ports. Options are: AUTO, COM1/3F8, COM2/2F8, COM3/3E8, COM4/2E8, and DISABLED. |
| **Onboard Parallel Port** | Select a logical LPT port address and corresponding interrupt for the physical parallel port. Options are: DISABLED, 278/IRQ5, 3BC/IRQ7, and 378/IRQ7. |
| **Parallel Port Mode** | Select an operating mode for the onboard parallel (printer) port. Select NORMAL unless you are certain your hardware and software both support one of the other available modes. Options are: NORMAL, EPP, ECP, ECP+EPP. |



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

*This page is intentionally blank.*



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

# CHAPTER 5 Board Resources

## Chapter Scope

This chapter describes the software- and hardware-configurable options of the V5B, as well as interrupts, DMA channels, and non-AT registers.

## V5B Options

### Processor Type & Speed

The V5B is designed to accommodate several Pentium processors at a variety of speeds. It can operate with a 133MHz, 166MHz, 200 MHz and 233MHz Pentium MMX processor. The AMD K6-2 is also available in speeds of 266MHz, 300MHz and 333MHz.

### DRAM Size

The V5B can accommodate varying sizes of several different memory technologies through the use of Quadword Memory (QM) modules. The QM modules are available in 32, 64, 128 and 256 megabyte capacities. The technologies available are Fast Page and EDO (extended data out).

## Video RAM

The V5B comes standard with two megabytes of video memory. Table 5-1 shows the colors available at each standard resolution.

| Resolution | Colors |
|---|---|
| 640×480 | 16.7 Million |
| 800×600 | 16.7 Million |
| 1024×768 | 65 Thousand |
| 1280×1024 | 256 |

Table 5-1 Simultaneous Colors Available at Standard Resolutions



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

## Front Panel LEDs

On the front panel of the V5B there are six LEDs which provide important information regarding the status of the board. These LEDs are, from top to bottom:

**SCON** This LED is ON when the V5B is configured as the System Controller in a VMEbus system. This is an important indicator as it allows you to more easily verify that you have only one System Controller on the VMEbus.

**RUN** This LED indicates that the processor is active. This LED may flicker as other devices occupy the local bus, but extended outage could indicate that the board is locked up.

**ENET** This LED indicates Ethernet network activity. Note that it is not limited to packets intended for or sent from the V5B. Rather, it indicates any network packets, regardless of source or destination.

**VMA** This LED is illuminated when the V5B is performing a master access to the VMEbus.

**VSA** This LED is illuminated when the V5B is being accessed as a slave from the VMEbus.

**SYS** This LED is user-defined and under user software control through the User Register (see User Register in the programming section).



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

## PC Interrupts

The 17 PC interrupts (including NMI) are allocated as shown in Table 5-2. Note that these are the default allocations, some can be changed. The PCI Interrupt is selectable between IRQ5, IRQ10, IRQ11, and IRQ15. IRQ11 is the default.

| Interrupt | Interrupt Number | Description/Use |
|---|---|---|
| NMI | N/A | Parity Error |
| IRQ0 | 8 | Timer |
| IRQ1 | 9 | Keyboard |
| IRQ2 | A | Chain to Interrupts 8–15 |
| IRQ3 | B | COM2 |
| IRQ4 | C | COM1 |
| IRQ5 | D | Available; Selectable as PCI Interrupt** |
| IRQ6 | E | Floppy Disk* |
| IRQ7 | F | LPT1 |
| IRQ8 | 70 | Real Time Clock |
| IRQ9 | 71 | Re-Directed to IRQ2 |
| IRQ10 | 72 | Default PCI Interrupt D (Ethernet)** |
| IRQ11 | 73 | Default PCI Interrupt A (SCSI)** |
| IRQ12 | 74 | Bus Mouse |
| IRQ13 | 75 | Coprocessor |
| IRQ14 | 76 | IDE Hard Disk* |
| IRQ15 | 77 | Available; Selectable as PCI Interrupt** |

Table 5-2 PC Interrupt Allocation

* These devices can be disabled to free the occupied interrupt.

** The PCI Interrupt is configurable as PC interrupt IRQ5, IRQ10, IRQ11, or IRQ15. IRQ11 is the default setting for INTA and IRQ10 is the default for INTD. Any of these PC interrupts to which the PCI Interrupt is not assigned can be considered available for other uses.

## DMA Channels

The eight PC DMA channels are allocated as shown in Table 5-3. These are the default allocations, some can be changed. With all onboard devices enabled, six DMA channels are available for use by peripheral cards. Disabling the Floppy Disk will free DMA channel 2.

| DMA Channel | Description/Use |
|---|---|
| 0 | Memory Refresh |
| 1 | Unassigned, Available |
| 2 | Floppy Disk* |
| 3 | Unassigned, Available |
| 4 | Unassigned, Available |
| 5 | Unassigned, Available |
| 6 | Unassigned, Available |
| 7 | Unassigned, Available |

Table 5-3 DMA Channel Allocation

* These devices can be disabled to free the occupied DMA channel.

## V5B Registers

The V5B has several non-AT registers such as Reset Control Register and the Serial EEPROM which allow complete control of the hardware by software. These registers are fully documented in the Programming section.



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

Figure 5-1 V5B Memory Map

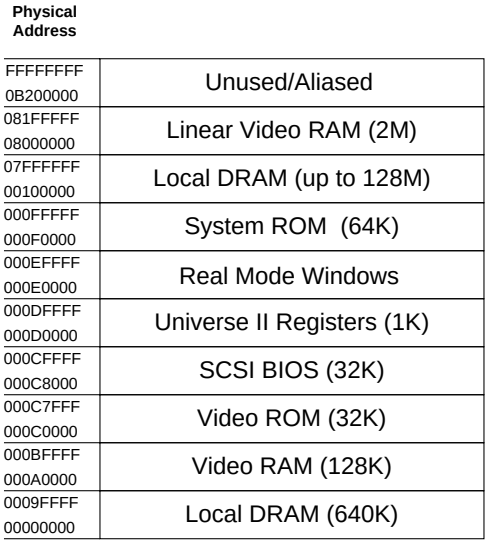
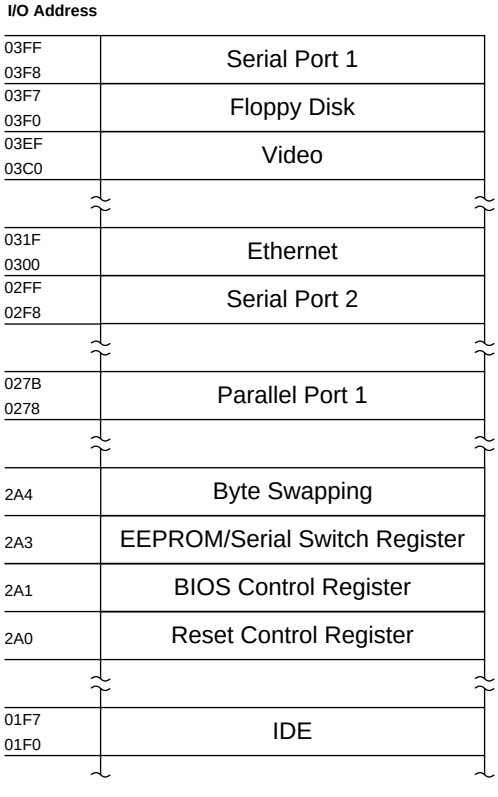



Figure 5-2 V5B I/O Map




Note that Local DRAM entries are dependent on the amount of DRAM installed. Also, some devices such as Floppy can be disabled, freeing space in the I/O map.



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

| Video Mode | Mode Type | Display Adapter | Pixel Resolution | Font Size | Chars | Colors | Dot Clock (MHz) | Horiz. Freq. (MHz) | Vert. Freq. (Hz) | Video Mem. (KB) | CRT |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 00h | Text | CGA | 320´200 | 8´8 | 40´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| | | EGA[2] | 320´350 | 8´14 | 40´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| | | VGA[1] | 360´400 | 9´16 | 40´25 | 16 | 28 | 31.5 | 70 | 256 | A,B,C |
| 01h | Text | CGA | 320´200 | 8´8 | 40´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| | | EGA[2] | 320´350 | 8´14 | 40´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| | | VGA[1] | 360´400 | 9´16 | 40´25 | 16 | 28 | 31.5 | 70 | 256 | A,B,C |
| 02h | Text | CGA | 640´200 | 8´8 | 80´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| | | EGA[2] | 640´350 | 8´14 | 80´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| | | VGA[1] | 720´400 | 9´16 | 80´25 | 16 | 28 | 31.5 | 70 | 256 | A,B,C |
| 03h | Text | CGA | 640´400 | 8´8 | 80´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| | | EGA[2] | 640´350 | 8´14 | 80´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| | | VGA[1] | 720´400 | 9´16 | 80´25 | 16 | 28 | 31.5 | 70 | 256 | A,B,C |
| 04h | Graph | All | 320´200 | 8´8 | 40´25 | 4 | 25 | 31.5 | 70 | 256 | A,B,C |
| 05h | Graph | CGA | 320´200 | 8´8 | 40´25 | 4 | 25 | 31.5 | 70 | 256 | A,B,C |
| | | EGA | 320´200 | 8´8 | 40´25 | 4 | 25 | 31.5 | 70 | 256 | A,B,C |
| | | VGA | 320´200 | 8´8 | 40´25 | 4 | 25 | 31.5 | 70 | 256 | A,B,C |
| 06h | Graph | All | 640´200 | 8´8 | 80´25 | 2 | 25 | 31.5 | 70 | 256 | A,B,C |
| 07h | Text | MDA | 720´350 | 9´14 | 80´25 | Mono | 28 | 31.5 | 70 | 256 | A,B,C |
| | | EGA | 720´350 | 9´14 | 80´25 | Mono | 28 | 31.5 | 70 | 256 | A,B,C |
| | | VGA | 720´400 | 9´16 | 80´25 | Mono | 28 | 31.5 | 70 | 256 | A,B,C |
| 08hÐ0Ch | | Reserved | | | | | | | | | |
| 0Dh | Graph | E/VGA | 320´200 | 8´8 | 40´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| 0Eh | Graph | E/VGA | 640´200 | 8´8 | 80´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| 0Fh | Graph | E/VGA | 640´350 | 8´14 | 80´25 | Mono | 25 | 31.5 | 70 | 256 | A,B,C |
| 10h | Graph | E/VGA | 640´350 | 8´14 | 80´25 | 16 | 25 | 31.5 | 70 | 256 | A,B,C |
| 11h | Graph | VGA | 640´480 | 8´16 | 80´25 | 2 | 25 | 31.5 | 60 | 256 | A,B,C |
| 12h | Graph | VGA | 640´480 | 8´16 | 80´30 | 16 | 25 | 31.5 | 60 | 256 | A,B,C |
| 13h | Graph | VGA | 320´200 | 8´8 | 40´25 | 256 | 25 | 31.5 | 70 | 256 | A,B,C |
| **Chips Extended Video Modes** | | | | | | | | | | | |
| 60h | Text | VGA | 1056´400 | 8´16 | 132´25 | 16 | 40 | 30.5 | 67.5 | 256 | A,B,C |
| 61h | Text | VGA | 1056´400 | 8´8 | 132´50 | 16 | 40 | 30.5 | 67.5 | 256 | A,B,C |
| 6A,70h | Graph | VGA | 800´600 | 8´16 | 100´37 | 16 | 40 | 38 | 60.5 | 256 | B,C |
| 72h(NI) | Graph | VGA | 1024´768 | 8´16 | 128´48 | 16 | 65 | 48.5 | 60 | 512 | C |
| 72h(I) | Graph | VGA | 1024´768 | 8´16 | 128´48 | 16 | 45 | 35.5 | 86 | 512 | A,B,C |
| 78h | Graph | VGA | 640´400 | 8´16 | 80´25 | 256 | 25 | 31.5 | 70 | 256 | A,B,C |
| 79h | Graph | VGA | 640´480 | 8´16 | 80´30 | 256 | 25 | 31.5 | 60 | 512 | A,B,C |
| 7Ch | Graph | VGA | 800´600 | 8´16 | 100´37 | 256 | 36 | 36 | 57 | 512 | A,B,C |

[1] Enhanced VGA mode. Otherwise, the VGA can emulate either the CGA or the EGA characteristics of this mode.
[2] The availability of these modes is dependent upon hardware and software configuration.
I = Interlaced, NI = Non-Interlaced

A = PS/2 analog CRT monitor or equivalent.
B = Multifrequency CRT (NEC Multisync 3D or equivalent)
C = Nanao Flexscan 9070s, NEC Multisync 5D or equivalent CRT monitor.

Color Palette is 256K (262,144) colors.

Table 5-4 V5B VIdeo Modes



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

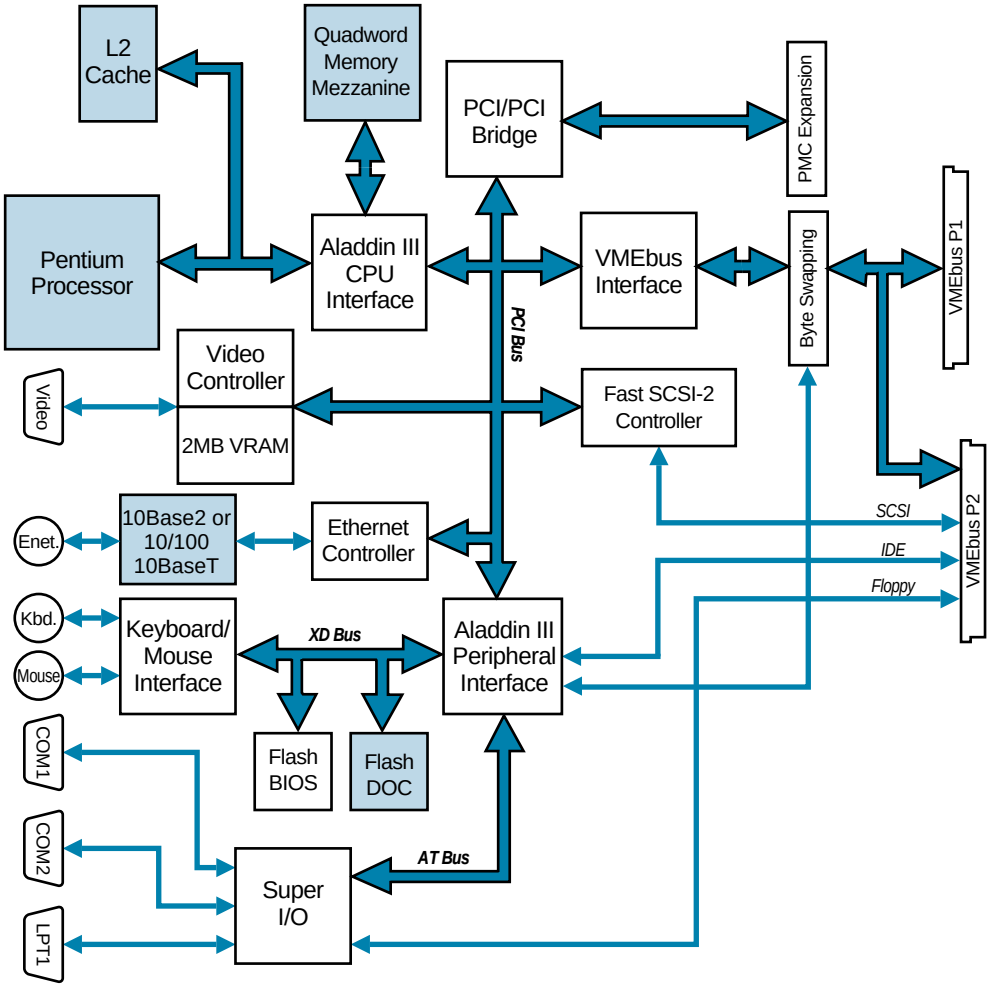



Figure 5-3 V5B Block Diagram



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

CHAPTER 6

# PC/AT Devices Programming

## Chapter Scope

This chapter describes programming the PC/AT devices of the V5B. The basic peripherals such as the serial ports, the parallel port, and the keyboard port are covered generally, as are the more advanced controllers for the video, floppy disk, and hard disk interfaces. Devices not usually found on a standard desktop PC/AT, such as onboard Ethernet and advanced SVGA video, are covered in more detail.

Note that it is beyond the scope of this manual to provide the programming details associated with the IBM PC/AT compatible hardware, MS-DOS software, and the ROM BIOS. Rather, it is suggested that one of the many good programming reference books for the PC/AT architecture be consulted. We highly recommend that you obtain a copy of *Advanced MS-DOS* which is published by Microsoft Press. It presents the details of the software environment of MS-DOS and the associated calls to the ROM BIOS. If you are planning to program in 80x86 assembler or a high-level language such as ‘C’ and need to access system resources directly, you will find this book absolutely essential.

## PC/AT Compatible ROM BIOS

The ROM BIOS is a firmware (installed in ROM) resident set of routines which is used to interface the particular system hardware to the MS-DOS operating system, and subsequently to MS-DOS applications. The use of a ROM BIOS in the original IBM PC/AT design has allowed programs to be developed specifically for the operating system environment without concern for the underlying hardware. Programs developed using the AT BIOS can be run on a variety of PC/AT compatible computers without modification. The BIOS provides a very rigid interface between the operating system and applications, and the hardware devices of the system. MS-DOS accesses devices through BIOS calls. These routines must perform their functions and return to the operating system after completion, without disturbing other system functions.



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

SBS Embedded Computers has developed the ROM BIOS for complete compatibility with IBM's BIOS, without infringing on IBM's copyright. Since the BIOS is completely compatible in operation with the original, we can assure you that programs developed correctly will run, unmodified on the V5B.

In order to use the routines contained in the AT ROM BIOS, as well as the VME BIOS, you will need to know how to make BIOS calls through software interrupts. This is described in detail in *Advanced MS-DOS*.

## Serial Communications Interface

The NS16450/16550-compatible UART serial controllers within the SMC-FDC37C665 fully implement both the standard PC/AT COM1 and COM2 serial ports. These devices normally appear at the standard addresses of 3F8H–3FFH for COM1 and 2F8H–2FFH for COM2. These addresses are software configurable. Standard input and output with the COM ports is accomplished with BIOS calls through interrupt 14H. These calls are fully described in Section 3 of Advanced MS-DOS.

The V5B incorporates a COM2 with additional functionality. COM2 can be configured for an RS-232, RS-422, or RS-485 driver type. The COM2 driver type can be configured in the Setup utility, as well as by using bit 7 of the Serial Switch (See Serial Switch below).

**WARNING:** Be sure that the type of driver selected matches the type of actual connection. The differences in driver voltages and currents between the driver types may cause damage to a connection of a different driver type. Changing the COM2 configuration while the COM2 port has a connection is not recommended.

## Parallel Port

The parallel port is also provided by the SMC-FDC37C665 and is typically used for interfacing to Centronics-compatible parallel printers. MS-DOS defines the device, PRN, as the normal printer output device of the system. Therefore, the programmer should not require access to the parallel port directly. Rather, printer services available through the AT BIOS at interrupt 17H should be used. AT BIOS services are fully described in Section 3 of Advanced MS-DOS. Also, be aware that the parallel port of the V5B is capable of SPP, ECP and EPP modes.

## Keyboard Port

The keyboard interface uses the AMI 8742 which is fully compatible with the keyboard used on the IBM PC/AT. Built into the AT BIOS are calls for keyboard services using interrupt 16H. Section 3 of Advanced MS-DOS describes these calls in detail.

## Real Time Clock

The Real-Time Clock of the V5B is equivalent to the Motorola 146818A and is contained in the DS12887 IC. The RTC occupies 3FH indexed locations and contains registers for clock timekeeping and control, as well as 16 bytes of scratchpad RAM (reserved for use by AT Setup). As long as battery power is provided, the clock will continue updating the date and time registers as well as preserve the contents of the scratchpad RAM.



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

Accessing the RTC is accomplished by first writing the index address to I/O port address 70H. The valid index address range is 0-7FH. The first 40H locations are used by the V5B, while the remaining 40H are available for general use. The register or RAM data can then either be written or read at I/O port address 71H. The specific use of clock registers is covered in the data sheet on the Motorola 146818A.

**Notice:** The RTC clock is a 100-year device. No hardware provisions exist to denote the rollover in the year 2000. An attempt to circumvent this problem has been implemented in the BIOS. However, advanced operating systems such as Windows NT do not use the BIOS to interface to the RTC. As such, they may display the year 2000 as the year 1900, etc.

## Non-Volatile RAM

The Non-Volatile RAM is actually the scratchpad RAM of the RTC discussed previously. This RAM is battery-backed and contains data used by AT Setup.

## Reset Control Register

The Reset Control Register is used to configure the sources and effects of reset conditions. The register is located at I/O address 2A0H.

| RES | RES | RES | FPSR | RES | RES | RES | IGSR |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Figure 6-1 Reset Control Register (I/O 2A0H)

**Bit 0: Ignore VME SYSRESET (R/W)**
This bit, when clear, causes the V5B to ignore incoming SYSRESET conditions from the VMEbus. This feature is handy when the V5B is being used as the development platform in a VME system. In such a system, the ability to reset the other VMEbus boards without resetting the V5B is helpful. When this bit is set, the V5B will reset upon assertion of the VMEbus SYSFAIL signal.

**Bits 1–3: Reserved**
These bits are reserved for future use by SBS. Reads are not guaranteed to return any particular value. To avoid incompatibilities with future hardware, it is recommended that you leave these bits unchanged during writes.

**Bit 4: Front Panel Reset to VME SYSRESET (R/W)**
This bit determines whether or not the front panel Reset switch of the V5B will cause a VMEbus SYSRESET. If this bit is set, the front panel Reset switch will assert the VMEbus SYSRESET signal as well as reset the V5B. If this bit is clear, the front panel Reset switch will not assert the VMEbus SYSRESET signal and will only reset the V5B.

**Bits 5–7: Reserved**
These bits are reserved for future use by SBS. Reads are not guaranteed to return any particular value. To avoid incompatibilities with future hardware, it is recommended that you leave these bits unchanged during writes.



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

## BIOS Control Register

The BIOS Control Register controls the write enables to both the onboard FLASH BIOS and the BIOS socket. The register is located at I/O address 2A1H.

| BSEL | SBWE | OBWE | RES | RES | RES | RSX | STRM |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Figure 6-2 BIOS Control Register (I/O 2A1H)

**Bit 0: SCSI Termination Enable**
This bit enables the on board active SCSI terminators. Setting this bit enables the terminators, while clearing it disables them. For system operation with the V5B as the first device in the SCSI chain this bit must be set. CAUTION: Clearing this bit with SCSI BIOS enabled and the V5B as the first device in the SCSI chain can cause board failure.

**Bit 1: RS232/RS422 Select**
This bit controls the selection of the COM2 serial port mode. Setting this bit enables RS422 operation while clearing this bit enables RS232 operation.

**Bits 2–4: Reserved**
This bit is reserved for future use by SBS Embedded Computers. Reads are not guaranteed to return any particular value. To avoid incompatibilities with future hardware, it is recommended that you leave this bit unchanged during writes.

**Bit 5: Onboard BIOS Write Enable (R/W)**
This bit enables programming the onboard FLASH BIOS. For normal operation, this bit should be clear so that the onboard BIOS is a read-only device. It should not be set by user programs. WARNING: Setting this bit makes the BIOS suceptable to data corruption.

**Bit 6: Socketed BIOS Write Enable (R/W)**
This bit enables programmingof a FLASH ROM in the BIOS socket. For normal operation, this bit should be clear so that any socketed ROM is a read-only device. It should not be set by user programs. WARNING: Setting this bit makes a FLASH device in the socket suceptable to data corruption.

**Bit 7: BIOS Select**
This bit controls the location of the chip select line for the bios. Setting this bit enables the chip select for the on board flash device, disabling the socketed chip select line. Clearing this bit enables the socketed flash chip select line while disabling the on board chip select line. NOTE: This bit's status is controlled by jumper J1 at reset and power-up. This bit is set by an uninstalled jumper and is cleared by an installed jumper. Be sure that a valid BIOS is in the socket before installing the jumper or else board failure will result.



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

## User Register

The User Register provides a bit for controlling the SYS LED on the front panel. The purpose of this LED is user-defined.

| SYS | Reserved | | | | | | |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Figure 6-3 User Register (I/O 2A2H)

**Bits 0–6: Reserved**
These bits are reserved for future use by SBS Embedded Computers. Reads are not guaranteed to return any particular value. To avoid incompatibilities with future hardware, it is recommended that you leave these bits unchanged during writes.

**Bit 7: SYS LED Control (R/W)**
This bit, when set, illuminates the SYS LED on the front panel of the V5B. The purpose of this indicator is completely user-defined. Note that readbacks of this bit will return the opposite value of the value written.

## Serial EEPROM Register

The Serial EEPROM Register provides a direct interface to the program, data, and clock lines of the serially-programmable EEPROM (see Serial EEPROM below). Programming of this device requires knowledge of device-specific timing requirements, and should be done cautiously. Refer to the National Semiconductor*NM93C46 User's Manual* if you need to access this part directly.

The Serial EPROM Register is an eight-bit register at I/O address 2A3H. The bits are defined as follows:

| RSV | OEECS | EEDIN | RSV | RSV | UEECS | SDO | CLK |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

Figure 6-4 Serial EEPROM Register (I/O 2A3H)

**Bit 0: Serial Clock (R/W)**
This bit is directly connected to the SK Serial Data Clock line of the NM93C46 Serial EEPROM. It is used for clocking serial data transfers to and from the Serial EEPROM.

**Bit 1: Serial Data Out (R/W)**
This bit is directly connected to the DI pin of the NM93C46. It is used to send data to the Serial EEPROM.

**Bit 5: Serial EEPROM Data In (R)**
This bit is directly connected to the DO pin of the NM93C64 during reads. It is used to receive serial data from the serial EEPROM.



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

**Bit 6: Serial EEPROM Chip Select (R/W)**
This bit is directly connected to the CS pin of the NM93C46. It selects the NM93C46 chip, making it active and ready for read or write cycles.

**Bit 7: Reserved**
This bit is reserved for future use by SBS Embedded Computers. Reads are not guaranteed to return any particular value. To avoid incompatibilities with future hardware, it is recommended that you leave this bit unchanged during writes.

## Serial EEPROM

The V5B incorporates a 64x16 bit serially-accessed Electrically Erasable Programmable Read Only Memory, or EPROM. This part is the National Semiconductor NM93C46, and is used for storing board setup and configuration data for SETV5B and the VME BIOS. Programming the EEPROM is facilitated through the Serial EPROM Register (see Serial EPROM Register above). However, user programs should make use of the EEPROM only under special circumstances. SBS Embedded Computers can provide you with more information about programming the EEPROM, as well as the dangers of doing so.

The EEPROM is 64 words in size. The first 43 words are used for VME setup values and board setup information. The remaining 21 words are available for general use.

Note: Like most EEPROM devices, the NM93C46 has a limited lifetime for write cycles. For the NM93C46, the average device life is $10^6$ write cycles. Conservative use of the EEPROM will ensure that it lasts the life of the board.

## Byte Swapping Control Register

**Byte Swapping Control register**
The Byte Swapping control register is used to configure the operation of the hardware byte swapping circuitry. The register is located at I/O address 2A4.

| RES | RES | RES | STAT | BYM1 | BYM0 | BYS1 | BYS0 |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| | Master | | Slave | |
|---|---|---|---|---|
| | **BYM1** | **BYM0** | **BYS1** | **BYS0** |
| M32 | 0 | 1 | 0 | 1 |
| I32 | 0 | 0 | 0 | 0 |
| I16 | 1 | 0 | 1 | 0 |

| Bit | Description |
|---|---|
| 0-1 | These bits control the byte swap mode when doing a VME Slave transfer. |
| 2-3 | These bits control the byte swap mode when doing a VME Master transfer. |
| 4 | If bits 0-3 are modified, the new mode is not in effect until the VMEbus is inactive. Bit 4 of the byte swapping register is read-only and indicates when the mode specified by by bits 0-3 is in effect. |

## Multi-Mode Video Controller and BIOS

The multi-mode video controller is implemented with a complex Large Scale Integration (LSI) device, the Cirrus Logic CL-GD5436. Primarily, you should not need to program the video controller directly, because the Video BIOS provides functions for video operation through interrupt 10H. These are fully defined in Section 3 of Advanced MS-DOS. However, when it is necessary to program the controller directly, refer to the Cirrus Logic CL-GD5436 Data Sheet. Refer toTable 5-4 on Page 5-5 (Last page of previous section) for the video modes supported by the V5B.



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

# CHAPTER 7 Byte Swapping

## Byte Swapping Modes

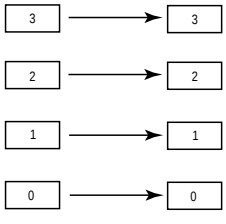
**Byte Swapping**

The V5B has Byte Swapping implemented in hardware. This allows for proper data transfer between the PCI and VME busses. The hardware implementation is much faster than a software implementation since no extra processor time is devoted to this task. Supported modes are: No swapping (M32), 32 bit swapping (I32), and 16 bit swapping (I16). Note that byte swapping is not applied to status ID's associated with interrupt processing.

| Type of VMEbus Access | Byte Swapping Modes | | |
|---|---|---|---|
| | M32 | I32 | I16 |
| Byte | Valid | Invalid | Invalid |
| Word | Valid | Invalid | Valid |
| Long Word | Valid | Valid | Invalid |

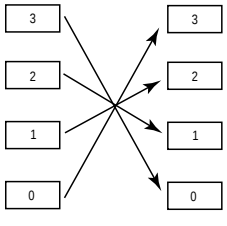
## Mode M32

M32 allows data to transfer with no swapping of the bytes. All address and data modes are supported with M32 since there is no swapping of bytes. M32 is the default setting in the BIOS setup.




## Mode I32

I32 allows conversion of big-endian to little-endian on a 32-bit transfer. Supported address modes are A16, A24, and A32. Supported data mode is D32. As can be seen from the diagram on a four-byte transfer, bytes 0 and 3 are swapped, and bytes 1 and 2 are swapped.

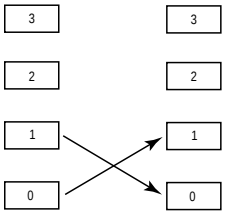





Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

## Mode I16

I16 allows conversion of big-endian to little-endian on a 16-bit transfer. Supported address modes are A16, A24, and A32. Supported data mode is D16. As can be seen from the diagram on two-byte transfer, bytes 0 and 1 are swapped.




The recommended method of controlling the byte swap mode is through the BIOS Setup program. Under the VME system menu both slave and master byte swapping can be set by the user. Another method of controlling byte swapping is through the register located in ISA space.

Byte swapping can be user controlled by a register in ISA space. This register is located at 2A4. Byte swap mode and bit settings are shown in the table below. By changing the bits in this register the user can change the mode of byte swapping. Note that the byte swapping mode will not be changed until the VME bus is inactive. This prevents the byte swap mode from changing while a transfer is in progress.

## Byte Swapping Mode By Control Register

| RES | RES | RES | STAT | BYM1 | BYM0 | BYS1 | BYS0 |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

**Byte Swapping Control Register**

The Byte Swapping control register is used to configure the operation of the hardware byte swapping circuitry. The register is located at I/O address 2A4.

| | Master | | Slave | |
|---|---|---|---|---|
| | **BYM1** | **BYM0** | **BYS1** | **BYS0** |
| M32 | 0 | 1 | 0 | 1 |
| I32 | 0 | 0 | 0 | 0 |
| I16 | 1 | 0 | 1 | 0 |

| **Bit** | **Description** |
|---|---|
| 0-1 | These bits control the byte swap mode when doing a VME Slave transfer. |
| 2-3 | These bits control the byte swap mode when doing a VME Master transfer. |
| 4 | If bits 0-3 are modified, the new mode is not in effect until the VMEbus is inactive. Bit 4 of the byte swapping register is read-only and indicates when the mode specified by by bits 0-3 is in effect. |



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

CHAPTER 8

# VMEbus Slave Programming

## Chapter Scope

This chapter describes programming the VMEbus slave interface of the V5B. Configuring the V5B as a slave, using the slave access paths, and slave registers of the Universe II VMEbus Interface Controller are covered.

## VME Access Paths

The VME interface of the V5B provides four discrete paths for VMEbus data transfers. These “access paths” can be configured as either VMEbus slave access paths or VMEbus master access paths.

## VMEbus Slave Interface

The VMEbus Slave Interface of the V5B is provided by the Tundra Universe II VMEbus Interface Controller. The Universe II presents access paths to and from VME in terms of PCI and VME “images”. If an access path is configured as a slave path, then in terms of Universe II functions, the path consists of a VMEbus Slave Image, a PCI Master Image, and a VMEbus Slave Channel. The Universe II Block Diagram shown in Appendix A  illustrates these function blocks.

## VME Slave & PCI Master

In order for the V5B to be a VMEbus slave, the Universe II VMEbus Interface Controller has to be a PCI bus master. As such, the remainder of this chapter is divided into two related sections, Universe II as a VMEbus Slave, and Universe II as a PCI Master.

## Universe II as a VMEbus Slave

The Universe II VME Slave Channel accepts all of the addressing and data transfer modes documented in the VME64 specification (except A64 and those intended to support 3U applications, i.e. A40 and MD32). Incoming write transactions from the VMEbus may be treated as either coupled or posted, depending upon the programming of the VMEbus slave image (see “VME Slave Images” on page 8-8). With posted write transactions, data is written to a Posted Write Receive FIFO (RXFIFO), and the VMEbus master receives data



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

acknowledgment from the Universe II. Write data is transferred to the PCI resource from the RXFIFO without the involvement of the initiating VMEbus master (see “Posted Writes” on page 8-3 for a full explanation of this operation). With a coupled cycle, the VMEbus master only receives data acknowledgment when the transaction is complete on the PCI bus. This means that the VMEbus is unavailable to other masters while the PCI bus transaction is executed.

Read transactions may be either prefetched or coupled. If enabled by the user, a prefetched read is initiated when a VMEbus master requests a block read transaction (BLT or MBLT) and this mode is enabled. When the Universe II receives the block read request, it begins to fill its Read Data FIFO (RDFIFO) using burst transactions from the PCI resource. The initiating VMEbus master then acquires its block read data from the RDFIFO rather than from the PCI resources directly.

The Universe II becomes a VMEbus slave when one of its eight programmed slave images or register images are accessed by a VMEbus master (note that the Universe II cannot reflect a cycle on the VMEbus and access itself). Depending upon the programming of the slave image, different possible transaction types result (see “VME Slave Images” on page 8-8 for a description of the types of accesses to which the Universe II responds).

For reads, the transaction can be coupled or prefetched. Similarly, write transactions can be coupled or posted. The type of read or write transaction allowed by the slave image is dependent upon the programming of that particular VME slave image (see Figure 8-1 below and “VME Slave Images” on page 8-8). To ensure sequential consistency, prefetched reads, coupled reads, and coupled write operations are only processed once all previously posted write operations have completed (i.e. the RXFIFO is empty).


Figure 8-1 VMEbus Slave Channel Dataflow



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

Incoming cycles from the VMEbus can have data widths of 8-bits, 16-bits, 32-bits, and 64-bits. Although the PCI bus supports only two port sizes (32-bits and 64-bits), the byte lanes on the PCI bus can be individually enabled, which allows each type of VMEbus transaction to be directly mapped to the PCI databus.

In order for a VMEbus slave image to respond to an incoming cycle, the PCI master interface must be enabled (bit BM in the PCI_CSR register, Universe II Register Definitions, Table A.3).

## Coupled Transfers

A coupled transfer means that no FIFO is involved in the transaction and handshakes are relayed directly through the Universe II. Coupled mode is the default setting for the VMEbus slave images. Coupled transfers only proceed once all posted write entries in the RXFIFO have completed (see “Posted Writes” below).

A coupled cycle with multiple data beats (i.e. block transfers) on the VMEbus side is always mapped to single data beat transactions on the PCI bus, where each data beat on the VMEbus is mapped to a single data beat transaction on the PCI bus regardless of data beat size. No packing or unpacking is performed. The only exception to this is when a D64 VMEbus transaction is mapped to D32 on the PCI bus. The data width of the PCI bus is dependent upon the programming of the VMEbus slave image (32-bit or 64-bit, see “VME Slave Images” on page 8-8). The Universe II enables the appropriate byte lanes on the PCI bus as required by the VMEbus transaction. For example, a VMEbus slave image programmed to generate 32-bit transactions on the PCI bus is accessed by a VMEbus D08 BLT read transaction (prefetching is not enabled in this slave image). The transaction is mapped to single data beat 32-bit transfers on the PCI bus with only one byte lane enabled.

The Universe II does not generate RETRY* on the VMEbus, so a target-retry from a PCI slave will not be communicated to the VMEbus master. PCI transactions terminated with target abort or master abort are terminated on the VMEbus with BERR*. Note that the Universe II sets the R_TA or R_MA bits in the PCI_CS register (Universe II Register Definitions, Table A.3) when it receives a target abort or master abort.

## Posted Writes

A posted write involves the VMEbus master writing data into the Universe II’s RXFIFO, rather than directly to the PCI address. Write transactions from the VMEbus are processed as posted if the PWEN bit is set in the VMEbus slave image control register (see “VME Slave Images” on page 8-8). If the bit is cleared (the default setting) the transaction bypasses the FIFO and is performed as a coupled transfer (see above). Incoming posted writes from the VMEbus are queued in a 16-entry deep RXFIFO. Each entry in the RXFIFO can contain 64 address bits (with extra bits provided in an address entry for command information), or 64 data bits. Each incoming VMEbus address phase, whether it is 16-bit, 24-bit, or 32-bit, constitutes a single entry in the RXFIFO and is



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

followed by subsequent data entries. The address entry contains the translated PCI address space and command information mapping relevant to the particular VMEbus slave image that has been accessed (see “VME Slave Images” on page 8-8). For this reason, any re-programming of VMEbus slave image attributes will only be reflected in RXFIFO entries queued after the re-programming. Transactions queued before the re-programming are delivered to the PCI bus with the VMEbus slave image attributes that were in use before the re-programming.

Incoming non-block write transactions from the VMEbus require two entries in the RXFIFO: one address entry (with accompanying command information) and one data entry. The size of the data entry corresponds to the data width of the VMEbus transfer. Block transfers require at least two entries: one entry for address and command information, and one or more data entries. The VME Slave Channel packs data received during block transfers to the full 64-bit width of the RXFIFO. For example, a ten data phase D16 BLT transfer (20 bytes in total) does not require ten data entries in the RXFIFO. Instead, eight of the ten data phases (16 bits per data phase for a total of 128 bits) are packed into two 64-bit data entries in the RXFIFO. The final two data phases (32 bits combined) are queued in the next RXFIFO entry. When you add the address entry to the three data entries, this VMEbus block write has been stored in a total of four RXFIFO entries.

Unlike the PCI Slave Channel (see Appendix B), the VME Slave Channel does not retry the VMEbus if the RXFIFO does not have enough space to hold an incoming VMEbus write transaction. Instead, the DTACK* response from the VME Slave Interface is delayed until space becomes available in the RXFIFO. Since single transfers require two entries in the RXFIFO, two entries must be freed up before the VME Slave Interface asserts DTACK*. Similarly, the VME Slave Channel requires two available RXFIFO entries before it can acknowledge the first data phase of a BLT or MBLT transfer (one entry for the address phase and one for the first data phase). If the RXFIFO has no available space for subsequent data phases in the block transfer, then the VME Slave Interface delays assertion of DTACK* until a single entry is available for the next data phase in the block transfer.

The availability of RXFIFO entries depends upon how many transactions are queued in the RXFIFO. The VME Slave Channel permits only four transactions queued in the RXFIFO at any one time. A transaction is a VMEbus transaction with an address phase and one or more data phases. For example a single word write is one transaction. If four single word writes are queued in the RXFIFO, then the RXFIFO is considered full. If four transactions are already queued in the RXFIFO, assertion of DTACK* is delayed until one transaction is delivered to the PCI bus.



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

The PCI Master Interface uses transactions queued in the RXFIFO to generate transactions on the PCI bus. No address phase deletion is performed, so the length of a transaction on the PCI bus corresponds to the length of the queued VMEbus transaction. Non-block transfers are generated on the PCI bus as single data beat transactions. Block transfers are generated as one or more burst transactions, where the length of the burst transaction is programmed by the PABS bit in the MAST_CTL register.

The Universe II always packs or unpacks data from the VMEbus transaction to the PCI bus data width programmed into the VMEbus slave image (with all PCI bus byte lanes enabled). For example, consider a VMEbus slave image programmed for posted writes and a D32 PCI bus that is accessed with a VMEbus D16 block write transaction. The VMEbus D16 write transaction is mapped to a D32 write transaction on the PCI bus with all byte lanes enabled. (However, note that a single D16 transaction from the VMEbus is mapped to the PCI bus as D32 with only two byte lanes enabled).

During block transfers, the Universe II will pack data to the full negotiated width of the PCI bus. This may imply that for block transfers that begin or end on addresses not aligned to the PCI bus width different byte lanes may be enabled during each data beat.

If an error occurs during a posted write to the PCI bus, the Universe II uses the L_CMDERR register (Universe II Register Definitions, Table A.31) to log the command information for the transaction (CMDERR [3:0]). The L_CMDERR register also records if multiple errors have occurred (with the M_ERR bit) although the actual number is not given. The error log is qualified with the L_STAT bit. The address of the errored transaction is latched in the LAERR register (Table A.32). An interrupt is generated on the VMEbus and/or PCI bus depending upon whether the VERR and LERR interrupts are enabled.

### Prefetched Block Reads

Prefetching of read data occurs for VMEbus block transfers (BLT, MBLT) in those slave images that have the prefetch enable (PREN) bit set (see “VME Slave Images” on page 8-8). Without prefetching, block read transactions from a VMEbus master are handled by the VME Slave Channel as coupled reads. This means that each data phase of the block transfer is translated to a single data beat transaction on the PCI bus. In addition, only the amount of data requested during the relevant data phase is fetched from the PCI bus. For example, D16 block read transaction with 32 data phases on the VMEbus maps to 32 PCI bus transactions, where each PCI bus transaction has only two byte lanes enabled. Note the VMEbus lies idle during the arbitration time required for each PCI bus transaction, resulting in a considerable performance degradation.



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

With prefetching enabled, the VME Slave Channel uses a 16 entry deep RDFIFO to provide read data to the VMEbus with minimum latency. The RDFIFO is 64 bits wide, with additional bits for control information. If a VMEbus slave image is programmed for prefetching (see “VME Slave Images” on page 8-8), then a block read access to that image causes the VME Slave Channel to generate aligned burst read transactions on the PCI bus (the size of the burst read transactions is determined by the setting of the aligned burst size, PABS in the MAST_CTL register).These PCI burst read transaction are queued in the RDFIFO and the data is then delivered to the VMEbus. Note that the first data phase provided to the VMEbus master is essentially a coupled read, but subsequent data phases in the VMEbus block read are delivered from the RDFIFO and are essentially decoupled (see “Prefetched Reads” on page 8-2 for the impact on bus error handling).

The data width of the transaction on the PCI bus (32-bit or 64-bit) depends upon the setting of the LD64EN bit in the VME slave image control register (e.g. see Table A.60) and the capabilities of the accessed PCI slave. Internally, the prefetched read data is packed to 64 bits, regardless of the width of the PCI bus or the data width of the original VMEbus block read (no address information is stored with the data). Once one entry is queued in the RDFIFO, the VME Slave Interface delivers the data to the VMEbus, unpacking the data as necessary to fit with the data width of the original VMEbus block read (e.g. D16, or D32). The VME Slave Interface continuously delivers data from the RDFIFO to the VMEbus master performing the block read transaction. Because PCI bus data transfer rates exceed those of the VMEbus, it is unlikely that the RDFIFO will ever be unable to deliver data to the VMEbus master. For this reason, block read performance on the VMEbus will be similar to that observed with block writes. However, should the RDFIFO be unable to deliver data to the VMEbus master (which may happen if there is considerable traffic on the PCI bus or the PCI bus slave has a slow response) the VME Slave Interface delays DTACK* assertion until an entry is queued and is available for the VMEbus block read.

On the PCI side, prefetching continues as long as there is room for another transaction in the RDFIFO and the initiating VMEbus block read is still active. The space required in the RDFIFO for another PCI burst read transaction is determined by the setting of the PCI aligned burst size (PABS in the MAST_CTL register, Universe II Register Definitions, Table A.56). If PABS is set for 32 bytes, there must be four entries available in the RDFIFO; for aligned burst size set to 64 bytes, eight entries must be available. When there is insufficient room in the RDFIFO to hold another PCI burst read, the read transactions on the PCI bus are terminated and only resume if room becomes available for another aligned burst AND the original VMEbus block read is still active. When the VMEbus block transfer terminates, any remaining data in the RDFIFO is purged.



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

Regardless of the read request, the data width of prefetching on the PCI side is full width with all byte lanes enabled. If the request is unaligned, then the first PCI data beat will have only the relevant byte lanes enabled. Subsequent data beats will have full data width with all byte lanes enabled. If LD64EN is set in the VME Slave image, the Universe II requests D64 on the PCI bus by asserting REQ64# during the address phase. If the PCI slave does not respond with ACK64#, subsequent data beats are D32.

The Universe II only translates errors from the PCI bus to the VMEbus during the first data beat (the coupled portion) of the prefetched read. A target abort on the PCI bus is translated as a BERR* signal on the VMEbus. However, a target abort on subsequent data beats during a prefetched read is ignored, but does cause prefetching to stop. There is no logging of the error, and all previously fetched data is provided to the VMEbus master. Once the block read on the VMEbus extends beyond the available data in the RDFIFO, then a new prefetch is initiated on the PCI bus. The point at which the new prefetch is initiated corresponds to where the error previously occurred. If the error occurs again, it will be on the first (coupled) data beat of the transaction and will be translated to the VMEbus as a BERR* signal. As with all coupled errors, no error is logged and no interrupt is generated. It should be anticipated that two target aborts may be generated before the VMEbus block read is terminated with BERR*.

## VMEbus Lock Commands

The Universe II supports VMEbus lock commands as described in the VME64 specification. Under the specification, ADOH cycles are used to execute the lock command (with a special AM code). Any resource locked on the VMEbus cannot be accessed by any other resource during the bus tenure of the VMEbus master. If the Universe II receives a VMEbus lock command, it asserts LOCK# to the addressed resource on the PCI bus. The Universe II holds the PCI bus lock until the VMEbus lock command is terminated, i.e. when BBSY* is negated. All subsequent slave VMEbus transactions are coupled while the Universe II owns PCI LOCK#. Note that the VME Slave Channel has dedicated access to the PCI Master Interface during the locked transaction.

## VMEbus Read Modify Writes

A read-modify-write (RMW) cycle allows a VMEbus master to read from a VMEbus slave and then write to the same resource without relinquishing bus tenure between the two operations. Each of the Universe II slave images can be programmed to map RMW transactions to PCI locked transactions. If the LLRMW enable bit is set in the appropriate VMEbus slave image control register (e.g. Universe II Register Definitions, Table A.60), then every non-block slave read is mapped to a coupled PCI locked read. LOCK# will be held on the PCI bus until AS* is negated on the VMEbus. Every non-block slave read is assumed to be a RMW since there is no possible indication from the VMEbus master that the single cycle read is just a read or the beginning of a RMW.



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

If the LLRMW enable bit is not set and the Universe II receives a VME RMW cycle, the read and write portions of the cycle will be treated as independent transactions on the PCI bus: i.e., a read followed by a write. The write may be coupled or decoupled depending on the state of the PWEN bit in the accessed slave image.

Note: There may be a performance loss for reads that are processed through a RMW-capable slave image. Some of this comes about due to the sampling of and arbitration for LOCK# by the Universe II's PCI Master Interface. More of a performance loss can arise if LOCK# is currently owned by another PCI master.

## Register Accesses

The Universe II registers are initially setup to appear at local addresses D000:0000 to D000:FFFF. See Universe II Register Definitions, Page A-1, for a full description of register mapping and register access.

## DTACK Rescinding

The user can increase performance in many systems by programming the Universe II to rescind DTACK*. When the Universe II releases the VMEbus, it can be programmed to either just release DTACK* (in which case DTACK* floats to its negated level) or rescind DTACK* (in which case the Universe II actively drives DTACK* to its negated level). DTACK* rescinding is programmed by setting the RESCIND bit in the MISC_CTL register (Table A.57) and is enabled by default.

## VME Slave Images

The Universe II accepts accesses from the VMEbus within specific programmed slave images. Each VMEbus slave image opens a window to the resources of the PCI bus and, through its specific attributes, allows the user to control the type of access to those resources. The tables below describe programming for the VMEbus slave images by dividing them into VMEbus, PCI bus and Control fields.

| Field | Register Bits | Description |
|---|---|---|
| base | BS[31:12] in Table A.61 | multiples of 4 or 64 Kbytes (base to bound: maximum of 4 GBytes) |
| bound | BD[31:12] in Table A.62 | |
| address space | VAS in Table A.60 | A16, A24, A32, User 1, User 2 |
| mode | SUPER in Table A.60 | supervisor and/or non-privileged |
| type | PGM in Table A.60 | program and/or data |

Table 8-1 VMEbus Fields for VME Slave Image



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

| Field | Register Bits | Description |
|---|---|---|
| image enable | EN in Table A.60 | enable bit |
| posted write | PWEN in Table A.60 | posted write enable bit |
| prefetched read | PREN in Table A.60 | prefetched read enable bit |
| enable PCI D64 | LD64EN in Table A.60 | enables 64-bit PCI bus transactions |

Table 8-2 PCI Bus Fields for VME Slave Image

| Field | Register Bits | Description |
|---|---|---|
| translation offset | TO[31:12] in Table A.63 | offsets VMEbus slave address to a selected PCI address |
| address space | LAS in Table A.60 | Memory, I/O, Configuration |
| RMW | LLRMW in Table A.60 | RMW enable bit |

Table 8-3 Control Fields for VME Slave Image

Note: If the programming for an image is changed after the transaction is queued in the FIFO, the transaction's attributes are not changed. Only subsequent transactions are affected by the change in attributes.

## VMEbus Fields

Decoding for VMEbus accesses is based on the address, and address modifiers produced by the VMEbus master. Before responding to an external VMEbus master, the address must lie in the window defined by the base and bound addresses, and the Address Modifier must match one of those specified by the address space, mode, and type fields.

The Universe II's four VME slave images (images 0 to 3) are bounded by A32 space. The first of these (VME slave image 0) has a 4 Kbyte resolution while VME slave images 1 to 3 have 64-Kbyte resolution (maximum image size of 4 GBytes). Typically, image 0 would be used as an A16 image since it provides the finest granularity of the four images. Any slave image (0–3) that is enabled must have its corresponding master image disabled.

Note that the address space of a VMEbus slave image must not overlap with the address space for the Universe II's control and status registers. Since register accesses take priority, any access to the overlap region results in a register access.



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

## PCI Bus Fields

The PCI bus fields specify how the VMEbus transaction is mapped to the appropriate PCI bus transaction. The translation offset field allows the user to translate the VMEbus address to a different address on the PCI bus. The translation of VMEbus transactions beyond 4 Gbytes results in wrap-around to the low portion of the address range.

The PAS field controls generation of the PCI transaction command. The LLRMW bit allows indivisible mapping of incoming VMEbus RMW cycles to the PCI bus via the PCI LOCK# mechanism (see “VMEbus Read Modify Writes” on page 8-5). When the LLRMW bit is set, single cycle reads will always be mapped to single data beat locked PCI transactions. Setting this bit has no effect on non-block writes: they can be coupled or decoupled. However, note that only accesses to PCI Memory Space are decoupled, accesses to I/O or Configuration Space are always coupled.

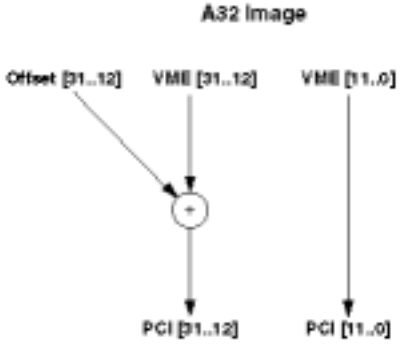



Figure 8-2 Address Translation for VMEbus to PCI Bus Transfers

## Control Fields

The control fields allow the user to enable a VMEbus slave image (using the EN bit), as well as specify how reads and writes will be processed. At power-up, all images are disabled and are configured for coupled reads and writes.

If the PREN bit is set, the Universe II will prefetch for incoming VMEbus block read cycles. It is the user's responsibility to ensure that prefetched reads are not destructive and that the entire image contains prefetchable resources.

If the PWEN bit is set, incoming write data from the VMEbus is loaded into the RXFIFO (see “Posted Writes” on page 8-3). Note that posted write transactions can only be mapped to Memory space on the PCI bus. Setting the PAS bit in the PCI fields to I/O or Configuration Space will force all incoming cycles to be coupled independent of this bit.



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

If the LD64EN bit is set, the Universe II will attempt to generate 64-bit transactions on the PCI bus by asserting REQ64#. The REQ64# line is asserted during the address phase in a 64-bit PCI system, and is the means of determining whether the PCI slave is a 64-bit port. If the slave asserts ACK64# with DEVSEL#, then the Universe II uses the 64-bit data bus. If the slave does not assert ACK64# with DEVSEL#, then the Universe II uses a 32-bit data bus. However, note that use of REQ64# requires extra clocks internally. Therefore, if no 64-bit targets are expected on the PCI bus then performance can be improved by disabling LD64EN on the VME slave images.

In order for a VMEbus slave image to respond to an incoming cycle, the PCI master interface must be enabled (bit BM in the PCI_CSR register, Table A.3).

## Configuring the VMEbus Slave using Setup

The VMEbus Slave Interface of the V5B can be configured using the embedded Setup utility. Figure 8-2 shows the VME Access Path 0 screen in Setup in one possible configuration. In this configuration, the slave is set to appear in VME A24 space. The Slave base address is set to 400000h and the bound address is set to 500000h. The PC address will be 400000h also, as the Translation Offset is set to 000000h. In this configuration, the V5B appears as an A24 slave at VME address 400000h which will decode to local address 400000h. Refer to Chapter 4 of this manual for more information on accessing and using the Setup utility.

## Example Slave Program

The example C program shown in Figure 8-3 demonstrates how to configure and enable the slave interface of the V5B.



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

```
#include <stdio.h>
#include <stdlib.h>
#include <dos.h>

#define      VERSION                    “0.90”

int image;                 // Image number
int os;                          // Image array offset

//
//  Configure the selected path as VME slave.
//

//  Initialize/Disable PCI Slave Image so Universe II can become VME Slave.
//
void u_initl( unsigned int lsi_base, unsigned int lsi_bd )
{
        unsigned char far *uregs;
        unsigned char bval=0;
        uregs = MK_FP( 0xD000, 0 );
        if( image == 0 ) {
        bval = uregs[0x105+os] & 0x0f;
        uregs[0x105+os] = bval | ((lsi_base & 0xf) << 4);// Set PCI mem base
        }
        uregs[0x106+os] = (lsi_base >> 4) & 0xff;
        uregs[0x107+os] = (lsi_base >> 12) & 0xff;
        if( image == 0 ) {
        bval = uregs[0x109+os] & 0x0f;
        uregs[0x109+os] = bval | ((lsi_bd & 0xf) << 4);// Set PCI mem bound
        }
        uregs[0x10a+os] = (lsi_bd >> 4) & 0xff;
        uregs[0x10b+os] = (lsi_bd >> 12) & 0xff;
        bval = uregs[0x102+os] & 0x3f;
        uregs[0x102+os] = bval | 0x80;            // vme max width to 32 bits
        uregs[0x100+os] &= 0xfc;                  // set for pci memory space
        bval = uregs[0x403] & 0x0f;
        uregs[0x403] = bval | 0x10;               // 64 retries
        uregs[0x406] &= ~(0x19);           // syscon, etc.
        uregs[0x103+os] &= ~(0x80);               // disable pci slave image
}


//  Enable and initialize VME Slave Image.
//
void u_initv( unsigned long vsi_base, unsigned long vsi_bd )
{
        unsigned char far *uregs;
        unsigned char bval=0;      uregs = MK_FP( 0xD000, 0 );
        if( image == 0 ) {
        bval = uregs[0xf05+os] & 0x0f;
        uregs[0xf05+os] = bval | ((vsi_base & 0xf) << 4);// Set VME mem base
        }
        uregs[0xf06+os] = (vsi_base >> 4) & 0xff;
        uregs[0xf07+os] = (vsi_base >> 12) & 0xff;
        if( image == 0 ) {
        bval = uregs[0xf09+os] & 0x0f;
```

Figure 8-3  Slave Access Example Program



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

```
        uregs[0xf09+os] = bval | ((vsi_bd & 0xf) << 4);// Set VME mem bound
        }
        uregs[0xf0a+os] = (vsi_bd >> 4) & 0xff;
        uregs[0xf0b+os] = (vsi_bd >> 12) & 0xff;
        uregs[0xf0d+os] &= 0x0f;
        uregs[0xf0e+os] = 0;                    // Translation Offset
        uregs[0xf0f+os] = 0;
        uregs[0xf02+os] |= 0x01;        // set for A24
        uregs[0xf03+os] |= 0xe0;        // Enable VME slave image 0
                                        // +posted write enable+read
                                        // prefetch
}


void main(void)
{
       int getch(void);
       void u_initl( unsigned int, unsigned int );
       void u_initv( unsigned long, unsigned long );
       unsigned int bc, i;
       unsigned char parms, ch;
       unsigned long va;
       unsigned char la[8];
       int s;

       printf("Set for VME Slave Operation - Version %4s\n", VERSION);
       printf("Select VME Slave Image (0 - 3)\n");
       ch = getch();
       switch (ch) {
              case '0':
                     image = 0;
                     break;
              case '1':
                     image = 1;
                     break;
              case '2':
                     image = 2;
                     break;
              case '3':
                     image = 3;
                     break;
              default:
                     printf("Illegal Entry\n");
                     exit(0);
       }
       os = image * 0x14;         // 14 hex bytes between images
       u_initl(0xe0, 0xf0);            // Disable + initialize PCI slave image
       u_initv(0x0l, 0xffffl);   // Enable + initialize VME slave image
       printf("VME Slave Image %1.1i Enabled\n", image);
}
```

Figure 8-3 Slave Access Example Program



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

*This page is intentionally blank.*



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

CHAPTER 9

# VMEbus Master Programming

## Scope

This chapter describes programming the VMEbus master interface of the V5B. Configuring the V5B as a VME master, using the four master images, and configuring the master registers of the Tundra Universe II chip are covered.

## VMEbus Master Interface

The VMEbus Master Interface of the V5B is provided by the Tundra Universe II VMEbus Interface Controller. As discussed in Chapter 8, the Universe II presents access paths to and from VME in terms of PCI and VME “images”. If an access path is configured as a master path, then in terms of Universe II functions, the path consists of a VMEbus Master Image, a PCI Slave Image, and a VMEbus Master Channel. The Universe II Block Diagram shown in Appendix A illustrates these function blocks. Whenever a VME master image is enabled, the corresponding slave image must be disabled.

## Master Access Paths

In Chapter 8 we discussed the four VME access paths provided by the V5B. Each of these paths can be setup to provide a local address range through which VME Master access can be made.

## Master Accesses in Real Mode

Any of the four access paths can be setup to provide a “Real Mode Window”, a range of address space on the V5B that provides a window to VMEbus address space for programs running in 80x86 Real Mode. It provides an easy, convenient method of accessing the VMEbus from the V5B running in 80x86 Real Mode. To use an access path to provide a Real Mode Window, configure the path with a base address of E0000 and a bound address of EFFFF. This local address range from E000:0000 to E000:FFFF will provides a 64 kilobyte page into VMEbus space. As such, it can provide a direct one-to-one map to VMEbus A16 (Short I/O) space. The Real Mode Window can be paged through VMEbus space for A24 and A32 accesses by changing the translation offset of the path.



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

**Note**: If the Real Mode Window is used, the E000  region must be excluded from any extended memory manager such as EMM386.EXE, and from Microsoft Windows. Also, be aware that calls to the Read VMEbus and Write VMEbus BIOS functions (functions 20 & 21) use the Real Mode Window E0000 region for VME access.

## The Universe II as VMEbus Master

The Universe II becomes VMEbus master when the VME Master Interface is internally requested by the PCI Bus Slave Channel, the DMA Channel, or the Interrupt Channel. The Interrupt Channel always has priority over the other two channels. Several mechanisms are available to configure the relative priority that the PCI Bus Slave Channel and DMA Channel have over ownership of the VMEbus Master Interface.

The Universe II's VME Master Interface generates all of the addressing and data transfer modes documented in the VME64 specification (except A64 and those intended to support 3U applications, i.e. A40 and MD32). The Universe II is also compatible with all VMEbus modules conforming to pre-VME64 specifications. As VMEbus master, the Universe II supports Read-Modify-Write (RMW), and Address-Only-with-Handshake (ADOH) but does not accept RETRY* as a termination from the VMEbus slave. The ADOH cycle is used to implement the VMEbus Lock command allowing a PCI master to lock VMEbus resources.

The Universe II becomes VMEbus master as a result of the following chain of events:

1. a PCI master accesses a Universe II PCI slave image (leading to VMEbus access) or the DMA Channel initiates a transaction,

2. either the Universe II PCI Slave Channel or the DMA Channel wins access to the VME Master Interface through internal arbitration, and

3. the Universe II Master Interface requests and obtains ownership of the VMEbus.

The Universe II will also become VMEbus master if the VMEbus ownership bit is set and in its role in VMEbus interrupt handling (see Chapter 10).

The following sections describe the function of the Universe II as a VMEbus master in terms of the different phases of a VMEbus transaction: addressing, data transfer, cycle termination, and bus release.



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

## Addressing Capabilities

Depending upon the programming of the PCI slave image (see “PCI Bus Slave Images” on page 9-5), the Universe II generates A16, A24, A32, and CR/CSR address phases on the VMEbus. The address mode and type (supervisor/ non-privileged and program/data) are also programmed through the PCI slave image. Address pipelining is provided except during MBLT cycles, where the VMEbus specification does not permit it.

The address and AM codes that are generated by the Universe II are functions of the PCI address and PCI slave image programming (see “PCI Bus Slave Images” on page 9-5) or through DMA programming.

The Universe II generates ADdress-Only-with-Handshake (ADOH) cycles in support of lock commands for A16, A24, and A32 spaces. ADOH cycles must be generated through the Special Cycle Generator.

To increase the flexibility of the Universe II’s address space programming, there are two User Defined AM codes that can be programmed through the USER_AM register (Universe II Register Definitions, Table A.59). After power-up, the two values in the USER_AM register default to the same VME64 user-defined AM code.

## Data Transfer Capabilities

The data transfer between the PCI bus and VMEbus is perhaps best explained by Figure 9-1. The Universe can be seen as a funnel where the mouth of the funnel is the data width of the PCI transaction. The end of the funnel is the maximum VMEbus data width programmed into the PCI slave image. For example, consider a 32bit PCI transaction accessing a PCI slave image with VDW set to 16 bits. A data beat with all byte lanes enabled will be broken into two 16-bit cycles on the VMEbus. If the PCI slave image is also programmed with block transfers enabled, the 32-bit PCI data beat will result in a D16 block transfer on the VMEbus. Write data is unpacked to the VMEbus and read data is packed to the PCI bus data width.

If the data width of the PCI data beat is the same as the maximum data width of the PCI slave image, then the Universe II maps the data beat to an equivalent VMEbus cycle. For example, consider a 32bit PCI transaction accessing a PCI slave image with VDW set to 32 bits. A data beat with all byte lanes enabled is translated to a single 32-bit cycle on the VMEbus.

As the general rule, if the PCI bus data width is less than the VMEbus data width then there is no packing or unpacking between the two buses. The only exception to this is during 32-bit PCI multi-data beat transactions to a PCI slave image programmed with maximum VMEbus data width of 64 bits. In this case, packing/unpacking occurs to make maximum use of the full bandwidth on both buses.



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

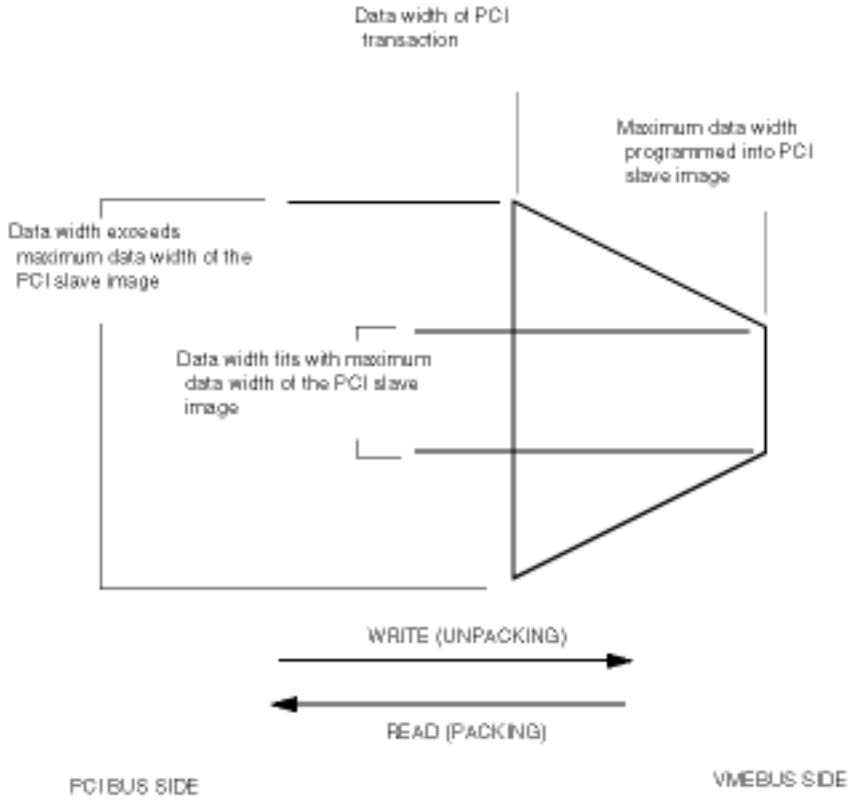



Figure 9-1 Transfer Width, Slave Image Width, & Data Packing/Unpacking

Only aligned VMEbus transactions are generated, so if the requested PCI data beat has unaligned or non-contiguous byte enables, then it is broken into multiple aligned VMEbus transactions no wider than the programmed VMEbus data width. For example, consider a three-byte PCI data beat (on a 32-bit PCI bus) accessing a PCI slave image with VDW set to 16 bits. The three-byte PCI data beat will be broken into two aligned VMEbus cycles: a single-byte cycle and a double-byte cycle (the ordering of the two cycles depends on the arrangement of the byte enables in the PCI data beat). If in the above example the PCI slave image has a VDW set to 8 bits, then the three-byte PCI data beat will be broken into three single-byte VMEbus cycles.

BLT/MBLT cycles are initiated on the VMEbus if the PCI slave image has been programmed with this capacity (see “PCI Bus Slave Images” on page 9-5). The length of the BLT/MBLT transactions on the VMEbus will be determined by the initiating PCI transaction or the setting of the PWON field in the MAST_CTL register (Universe II Register Definitions, Table A.56). For example, a single data beat PCI transaction queued in the TXFIFO results in a single data beat block transfer on the VMEbus. With the PWON field, the user can specify a transfer byte count that will be dequeued from the TXFIFO before the VME Master Interface relinquishes the VMEbus.



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

During DMA operations, the Universe II will attempt block transfers to the maximum length permitted by the VMEbus specification (256 bytes for BLT, 2 Kbytes for MBLT) and as limited by the VON counter.

## Cycle Terminations

The Universe II accepts BERR* or DTACK* as cycle terminations from the VMEbus slave. The assertion of BERR* indicates that some type of system error occurred and the transaction did not complete properly. A VMEbus BERR* received by the Universe II during a coupled transaction is communicated to the PCI master as a target abort. No information is logged if the Universe II receives BERR* in a coupled transaction. If an error occurs during a posted write to the VMEbus, the Universe II uses the V_AMERR register (Universe II Register Definitions, Table A.80) to log the AM code of the transaction (AMERR [5:0]), and the state of the IACK* signal (IACK bit, to indicate whether the error occurred during an IACK cycle). The current transaction in the FIFO is purged. The V_AMERR register also records if multiple errors have occurred (with the M_ERR bit), although the actual number of errors is not given. The error log is qualified by the value of the V_STAT bit. The address of the errored transaction is latched in the V_AERR register (Table A.81). When the Universe II receives a VMEbus error during a posted write, it generates an interrupt on the VMEbus and/or PCI bus depending upon whether the VERR and LERR interrupts are enabled (see “The Interrupt Channel” on Universe II Register Definitions, page A-52, Table A.43 and Table A.44).

DTACK* signals the successful completion of the transaction.

## PCI Bus Slave Images

The Universe II accepts accesses from the PCI bus with specific programmed PCI slave images. Each image opens a window to the resources of the VMEbus and allows the user to control the type of access to those resources.The tables below describe programming for the four standard PCI bus slave images (numbered 0 to 3) by dividing them into VMEbus, PCI bus and Control fields. One special PCI slave image separate from the four discussed below is described in “Special PCI Slave Image” on page 9-8.

| Field | Register Bits | Description |
|---|---|---|
| base | BS[31:12] in Table A.9 | multiples of 4 or 64 Kbytes (base to bound: maximum of 4 GBytes) |
| bound | BD[31:12] in Table A.10 | |
| address space | LAS in Table A.8 | Memory, I/O, Configuration |

Table 9-1 PCI Bus Fields for PCI Slave Image



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

| Field | Register Bits | Description |
|---|---|---|
| translation offset | TO[31:12] in Table A.11 | translates address supplied by PCI master to a specified VMEbus address |
| maximum data width | VDW in Table A.8 | 8, 16, 32, or 64 bits |
| address space | VAS in Table A.8 | A16, A24, A32, CR/CSR, User1, User2 |
| mode | SUPER in Table A.8 | supervisor or non-privileged |
| type | PGM in Table A.8 | program or data |
| cycle | VCT in Table A.8 | single or block |

Table 9-2 VMEbus Fields for PCI Slave Image

| Field | Register Bits | Description |
|---|---|---|
| image enable | EN in Table A.8 | enable bit |
| posted write | PWEN in Table A.8 | enable bit |

Table 9-3 Control Fields for PCI Slave Image

## PCI Bus Fields

All decoding for VMEbus accesses are based on the address and command information produced by a PCI bus master. The Universe II slave claims a cycle if there is an address match and if the command matches certain criteria.

All of the Universe II's four PCI slave images are A32-capable only. The first of these (PCI slave image 0) has a 4 Kbyte resolution while PCI slave images 1 to 3 have 64 Kbyte resolution. Typically, image 0 would be used for an A16 image since it has the finest granularity.

Note that the address space of a PCI bus slave image must not overlap with the address space for the Universe II's control and status registers. Since register accesses take priority, any access to the ovelap region results in a register access.



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

## VMEbus Fields

The VMEbus fields map PCI transactions to a VMEbus transaction, causing the Universe II to generate the appropriate VMEbus address, AM code, and cycle type. Some invalid combinations exist within the PCI slave image definition fields. For example, A16 and CR/CSR spaces do not support block transfers, and A16 space does not support 64-bit transactions. Note that the Universe II does not attempt to detect or prevent these invalid programmed combinations, and that use of these combinations may cause illegal activity on the VMEbus.

The 21-bit translation offset allows the user to translate the PCI address to a different address on the VMEbus. The figure below illustrates the translation process:.

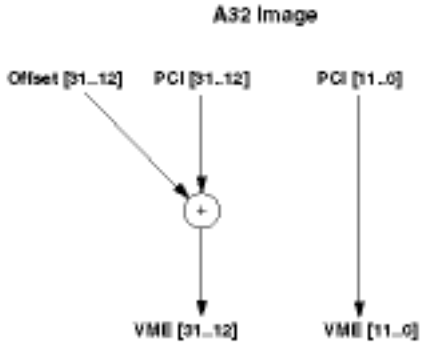



Figure 9-2 Address Translation for PCI Bus to VMEbus Transfers

The AM code generated by the Universe II is a function of: the VMEbus fields, and the data width and alignment generated by the PCI bus master. For RMW and ADOH cycles, the AM code also depends upon the settings for the Special Cycle Generator.

The address space, mode, type, and cycle fields control the VMEbus AM code for most transactions. Setting the cycle field to BLT enables the BLT cycle generation. MBLT cycles are generated when the maximum width is set to 64, the BLT bit is set, and the PCI master queues a transaction with at least 64 bits (aligned) of data.

The Universe II provides support for user defined AM codes. The USER_AM register (Universe II Register Definitions, Table A.59) contains AM codes identified as User1 and User2. If the user selects one of these two, then the corresponding AM code from the global register is generated on the VMEbus. This approach results in standard single cycle transfers to A32 VMEbus address space independent of other settings in the VMEbus fields.



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

## Control Fields

The control fields allow the user to enable a PCI slave image (the EN bit), as well as specify how writes are processed. If the PWEN bit is set, then the Universe II will perform posted writes when that particular PCI slave image is accessed. Posted write transactions are only decoded within PCI Memory space. Accesses from other spaces will result in coupled cycles independent of the setting of the PWEN bit.

## Special PCI Slave Image

The Universe II provides a special PCI slave image located in Memory, I/O, or Configuration space. Its base address is aligned to 64-Mbyte boundaries and its size is fixed at 64 Mbytes (decoded using PCI address lines [31:26]). The Special PCI Slave Image is divided into four 16Mbyte regions numbered 0 to 3 (see Figure 9-3 below). These separate regions are selected with PCI address bits AD [25:24]. For example, if AD [25:24 = 01, then region 1 is decoded. Within each region, the upper 64Kbytes map to VMEbus A16 space, while the remaining portion of the 16 Mbytes maps to VMEbus A24 space. Note that no offsets are provided, so address information from the PCI transaction is mapped directly to the VMEbus.

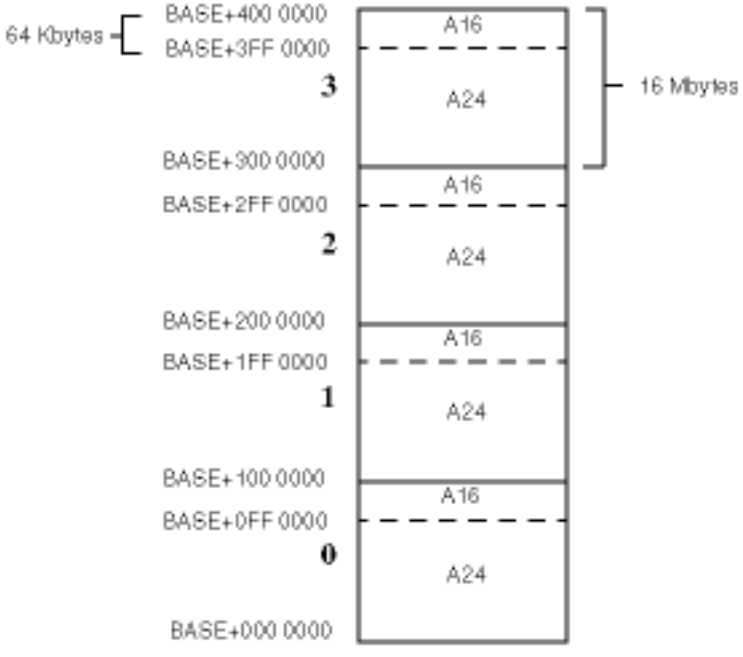



Figure 9-3 Memory Mapping in the Special PCI Slave Image

The general attributes of each region are programmed according to the tables below.



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

| Field | Register Bits | Description |
|---|---|---|
| base | BS[5:0] in Table A.30 | 64 Mbyte aligned base address for the image |
| address space | LAS [1:0] in Table A.30 | Places image in Memory, I/O, or Configuration space |

Table 9-4 PCI Bus Fields for the Special PCI Slave Image

| Field | Register Bits | Description |
|---|---|---|
| maximum data width | VDW in Table A.30 | separately sets each region for 16 or 32 bits |
| mode | SUPER in Table A.30 | separately sets each region as supervisor or non-privileged |
| type | PGM in Table A.30 | separately sets each region as program or data |

Table 9-5 VMEbus Fields for the Special PCI Slave Image

| Field | Register Bits | Description |
|---|---|---|
| image enable | EN in Table A.30 | enable bit for the image |
| posted write | PWEN in Table A.30 | enable bit for posted writes for the image |

Table 9-6 Control Fields for the Special PCI Slave Image

The special PCI slave image provides access to all of A16 and most of A24 space (all except the upper 64 Kbytes). By using the special PCI slave image for A16 and A24 transactions, it is possible to free the four standard PCI slave images (see “PCI Bus Slave Images” on page 9-5), which are typically programmed to access A32 space.

Note that some address space redundancy is provided in A16 space. The VMEbus specification requires only two A16 spaces, while the special PCI slave image allows for four A16 address spaces.

## VME Master Example Program

The C program shown in Figure 9-4 is an example of how to configure the VMEbus master interface and perform VME master operations using the V5B. The program allows the user to select one of the four VME access paths and perform read and write operations to the VMEbus through the selected path.



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

```
#include <time.h>
#include <stdio.h>
#include <stdlib.h>
#include <dos.h>

#define VERSION              “0.90”

int path;                    // Universe II path number
int os;                           // Universe II path offset
unsigned char vdw;                // VMEbus data width

//  funct20 - Read VMEbus.
//  ENTRY:    vadd = VMEbus address
//            bytes = number of bytes to transfer
//            padd = PCI bus (local) address
//            parms =
//            2:0    VME address space: 000=A16  001=A24  010=A32
//                                      101=CR/CSR  110=User1  111=User2
//            3      pgm/data: 0=data  1=program
//            4      mode: 0=non-privileged  1=supervisor
//
//  EXIT:     Nothing.
//
void funct20( unsigned long vadd, unsigned int bytes, unsigned char *padd,
                unsigned char parms)
{
       unsigned char bval=0;
       unsigned char btmp;
       unsigned char far *uregs;          // pointer to Univ reg space
       unsigned char far *vspace;         // pointer to local mem space
       unsigned long lval;
       unsigned int i;

       uregs = MK_FP( 0xd000, 0 );
       uregs[0x102+os] = (uregs[0x102+os] & 0xf8) | (parms & 0x07);
       bval = uregs[0x101+os] & 0x0e;           // isolate PGM,SUPER clear
VCT
       btmp = (parms >> 3) & 1;
       bval |= btmp << 6;                    // position pgm bit
       btmp = (parms >> 4) & 1;
       bval |= btmp << 4;                    // position super bit
       uregs[0x101+os] = bval;               // set them
       lval = (vadd >> 12) + 0xfff20;
       if( path == 0 ) {
       bval = uregs[0x10d] & 0xf;
       uregs[0x10d] = bval | ((lval & 0xf) << 4);
       }
       uregs[0x10e+os] = (lval >> 4) & 0xff;
       uregs[0x10f+os] = (lval >> 12) & 0xff;   // VME adr by Translation
Offset
       vspace = MK_FP( 0xe000,0 );        // VME space
       for( i=0; i<bytes; i++) {
            *padd = vspace[i + (vadd & 0xfff)];// VME to Local bus transfer
            padd++;
            }
}


//  funct21 - Write VMEbus.
//  ENTRY:    vadd = VMEbus address
```

Figure 9-4 Master Access Example



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

```
//            bytes = number of bytes to transfer
//            padd = PCI bus (local) address
//            parms =
//            2:0    VME address space: 000=A16  001=A24  010=A32
//                                      101=CR/CSR  110=User1  111=User2
//            3      pgm/data: 0=data  1=program
//            4      mode: 0=non-privileged  1=supervisor
//
//  EXIT:     Nothing.
//
void funct21( unsigned long vadd, unsigned int bytes, unsigned char *padd,
                 unsigned char parms)
{
       unsigned char bval, btmp;
       unsigned char far *uregs;          // pointer to Univ reg space
       unsigned char far *lspace;         // pointer to local mem space
       unsigned int i;
       unsigned long lval;

       uregs = MK_FP( 0xd000, 0 );
       uregs[0x102+os] = (uregs[0x102+os] & 0xf8) | (parms & 0x07);
       bval = uregs[0x101+os] & 0x0e;           // isolate PGM,SUPER clear
VCT
       btmp = (parms >> 3) & 1;
       bval |= btmp << 6;                     // position pgm bit
       btmp = (parms >> 4) & 1;
       bval |= btmp << 4;                     // position super bit
       uregs[0x101+os] = bval;                // set them
       lval = (vadd >> 12) + 0xfff20;
       if( path == 0) {
       bval = uregs[0x10d] & 0xf;
       uregs[0x10d] = bval | ((lval & 0xf) << 4);
       }
       uregs[0x10e+os] = (lval >> 4) & 0xff;
       uregs[0x10f+os] = (lval >> 12) & 0xff;   // VME address via T.O.
       lspace = MK_FP( 0xe000, 0 );           // VME space
       for( i=0; i<bytes; i++) {
            lspace[vadd] = *padd;           // Local bus to VMEbus xfer
            vadd++;
            padd++;
            }
}


//  Initialize the local (PCI) Slave Image so Universe II can become VME
Master
//  on Universe II Path 0, 1, 2, or 3.
//
//  Note: The local slave image registers are 14 hex bytes apart.
//
void u_init( unsigned int lsi_base, unsigned int lsi_bd )
{
       unsigned char far *uregs;          // pointer to Univ reg space
       unsigned char bval=0;
       uregs = MK_FP( 0xD000, 0 );        // Univ reg space=mem access
       if( path == 0 ) {
       bval = uregs[0x105] & 0x0f;
       uregs[0x105] = bval | ((lsi_base & 0xf) << 4);// Set PCI mem base
       }
       uregs[0x106+os] = (lsi_base >> 4) & 0xff;
       uregs[0x107+os] = (lsi_base >> 12) & 0xff;
```

Figure 9-4 Master Access Example



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

```
        if( path == 0 ) {
        bval = uregs[0x109] & 0x0f;
        uregs[0x109] = bval | ((lsi_bd & 0xf) << 4);// Set PCI mem bound
        }
        uregs[0x10a+os] = (lsi_bd >> 4) & 0xff;
        uregs[0x10b+os] = (lsi_bd >> 12) & 0xff;
        bval = uregs[0x102+os] & 0x3f;
        uregs[0x102+os] = bval | vdw;          // set VMEbus max data width
        uregs[0x100+os] &= 0xfc;               // set for pci memory space
        bval = uregs[0x403] & 0x0f;
        uregs[0x403] = bval | 0x10;            // 64 retries
        bval = uregs[0x406] & 0xe2;
        uregs[0x406] = bval | 0x04;        // rescind
        uregs[0xf03+os] &= ~(0x80);        // disable vme slave image
        uregs[0x103+os] |= 0x80;               // enable pci slave image
}


void main(void)
{
      int getch(void);
      clock_t clock(void);
      void u_init( unsigned int, unsigned int);
      void funct20( unsigned long, unsigned int, unsigned char *,
      unsigned char);
      void funct21( unsigned long, unsigned int, unsigned char *,
      unsigned char);
      unsigned int bc, lc, i, j, ecnt;
      unsigned char parms, ch;
      unsigned long va;
      clock_t start, fini;
      unsigned char la[4096], db[4096];
      int ddw, s;

      printf("VME Master Access Test - Version %4s\n", VERSION);
      printf("Select Path for VME Master Transfer(0-3):\n");
      ch = getch();
      switch(ch) {
            case '0':
                  path = 0;
                  break;
            case '1':
                  path = 1;
                  break;
            case '2':
                  path = 2;
                  break;
            case '3':
                  path = 3;
                  break;
            default:
                  printf("Illegal Entry\n");
                  exit(0);
            }
      printf("Using Path %1.1i\n", path);
      printf("0 for  8 bit VMEbus data width\n");
      printf("1 for 16 bit VMEbus data width\n");
      printf("2 for 32 bit VMEbus data width\n");
      printf("3 for 64 bit VMEbus data width\n");
      printf("Select width:\n");
      ch = getch();
      switch(ch) {
```

Figure 9-4 Master Access Example



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

```
        case ‘0’:
              vdw = 0 << 6;           // use 08-bit vme data width
              ddw = 8;
              break;
        case ‘1’:
              vdw = 1 << 6;           // use 16-bit vme data width
              ddw = 16;
              break;
        case ‘2’:
              vdw = 2 << 6;           // use 32-bit vme data width
              ddw = 32;
              break;
        case ‘3’:
              vdw = 3 << 6;           // use 64-bit vme data width
              ddw = 64;
              break;
        default:
              printf(“Illegal Entry\n”);
              exit(0);
        }
        printf(“Using %2.2i-bit data width\n”, ddw);
        os = path * 0x14;       // 14 hex bytes between images
        lc = 1024;              // loop count
        bc = 4096;                      // transfer byte count
        for( i=0; i<bc; i++ )         // initialize data buffer
              la[i] = i;
        u_init(0xe0, 0xf0);             // Enable + initialize pci slave 0
        parms = 0x11;                   // A24 + Data + Supervisor
        while(1) {
              printf(“\n0 = Quit\n”);
              printf(“1 = Write\n”);
              printf(“2 = Read\n”);
              printf(“3 = Compare Data\n”);
              printf(“Enter Option: \n”);
              ch = getch();
              switch (ch) {
              case ‘0’:         // End of program
                    printf(“Bye\n”);
                    exit(0);
              case ‘1’:         // VME Master write
                    printf(“Write VME address: “);
                    s = scanf(“%8lx”, &va);
                    if(s == 0) exit(0);
                    start = clock();
                    for( i=0; i<lc; i++ )
                          funct21( va, bc, la, parms);
                    fini = clock();
                    printf(“Wt time= %f\n”, (fini-start)/CLK_TCK);
                    break;
              case ‘2’:         // VME Master read
                    printf(“Read VME address: “);
                    s = scanf(“%8lx”, &va);
                    if(s == 0) exit(0);
                    start = clock();
                    for( i=0; i<lc; i++ )
                          funct20( va, bc, la, parms);
                    fini = clock();
                    printf(“Rd Time= %f\n”, (fini-start)/CLK_TCK);
                    printf(“ %2.2x %2.2x %2.2x %2.2x”, la[0], la[1], la[2],
la[3]);
                    printf(“ %2.2x %2.2x %2.2x %2.2x\n”, la[4], la[5], la[6],
```

Figure 9-4 Master Access Example



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

```
la[7]);
                  break;
            case '3':         // VME Master WT/RD/Compare
                  printf("Wt/Rd VME address: ");
                  s = scanf("%8lx", &va);
                  if(s == 0) exit(0);
                  for(i=0; i<bc; i++)
                        db[i] = i;  // pattern compare buffer
                  ecnt = 0;         // error count
                  for( i=0; i<lc; i++ ) {
                        funct21( va, bc, db, parms);
                        funct20( va, bc, la, parms);
                        for(j=0; j<bc; j++) {
                              if( la[j] != db[j] ) {
                                    printf("Got %2.2x, Expected %2.2x ",
la[j], db[j]);
                                    printf("at byte %4.4x of pass %4.4x\n",
j, i);
                                    ecnt++;
                                    }
                        }
                  }
                  printf("%i errors detected\n", ecnt);
                  break;
            default:
                  printf("Select 0,1 or 2.\n");
                  break;
            }  //End switch.
      }  //End while.
}  //End main.
```

Figure 9-4  Master Access Example



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

# CHAPTER 10 VMEbus Interrupt Programming

## Chapter Scope

This chapter covers programming the interrupt facilities of the V5B VMEbus interface. Generating VMEbus interrupts, handling VMEbus interrupts, and handling local VME-related interrupts are discussed.

## VME Interrups and the Combined PCI Interrupt

The Tundra Universe II provides a range of VMEbus-related interrupts for use in your application. All VMEbus interrupt sources are combined to form one PCI interrupt. This PCI interrupt can be mapped to one of several local IRQ lines. IRQ5, IRQ10, IRQ11, and IRQ15 can be selected, with IRQ11 as the default. To avoid conflict with other PCI devices (SCSI, Ethernet), a PCI steerable IRQ may also be used to exclusively handle Universe II interrupts. This PCI steerable interrupt can also be mapped to one of the local IRQ lines (5,10,11,15).

## Interrupter

The Universe II interrupt channel provides a flexible scheme to map interrupts to either the PCI bus or VMEbus interface. Interrupts are generated from either hardware or software sources (see “VMEbus Interrupt Generation” on page 10-5 for a full description of hardware and software sources). Interrupt sources can be mapped to any of the PCI bus or VMEbus interrupt output pins. Interrupt sources mapped to VMEbus interrupts are generated on the VMEbus interrupt output pins VIRQ# [7:1]. When a software and hardware source are assigned to the same VIRQn# pin, the software source always has higher priority.

Interrupt sources mapped to PCI bus interrupts are generated on one of the INT# [7:0] pins. To be fully PCI compliant, all interrupt sources must be routed to a single INT# pin.



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

For VMEbus interrupt outputs, the Universe II interrupter supplies an 8-bit STATUS/ID to a VMEbus interrupt handler during the IACK cycle, and optionally generates an internal interrupt to signal that the interrupt vector has been provided (see “VMEbus Interrupt Generation” on page 10-5).

Interrupts mapped to PCI bus outputs are serviced by the PCI interrupt controller. The CPU determines which interrupt sources are active by reading an interrupt status register in the Universe II. The source negates its interrupt when it has been serviced by the CPU (see “PCI Interrupt Generation” below).

## VMEbus Interrupt Handling

A VMEbus interrupt triggers the Universe II to generate a normal VMEbus IACK cycle and generate the specified interrupt output. When the IACK cycle is complete, the Universe II releases the VMEbus and the interrupt vector is read by the PCI resource servicing the interrupt output. Software interrupts are ROAK, while hardware, and internal interrupts are RORA.

## The Interrupt Channel

The Universe II Interrupt Channel is a module within the Universe II which handles prioritization and routing of various interrupt sources to interrupt outputs on the PCI bus and VMEbus. The interrupt sources are:

- the PCI LINT[7:0] lines,
- the VME IRQ*[7:1] lines,
- ACFAIL* and SYSFAIL*
- various internal signals

These sources can be routed to either the PCI LINT [7:0] lines or the VMEbus IRQ* [7:0] lines (see Figure 10-1 below). Each interrupt source is individually maskable and can be mapped to various interrupt outputs. Most interrupt sources can be mapped to one particular destination bus. The PCI sources, LINT[7:0], can only be mapped to the VMEbus interrupt outputs, while the VMEbus sources, VIRQ[7:1], can only be mapped to the PCI interrupt outputs. Some internal sources (for example, error conditions or DMA activity) can be mapped to either bus.

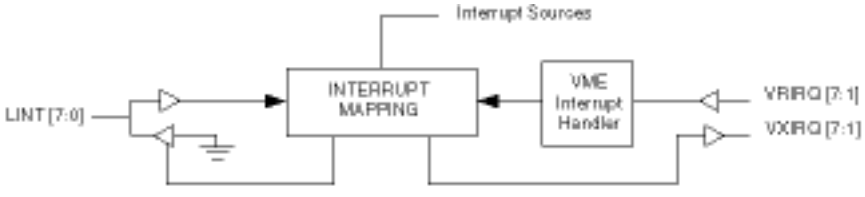



Figure 10-1 Interrupt Sources and Destinations



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

## PCI Interrupt Generation

The Universe II expands on the basic PCI specification which permits “single function” devices to assert only a single interrupt line. Eight PCI interrupt outputs are provided to give maximum flexibility, although if full PCI compliancy is required, the user may route all interrupt sources to a single PCI interrupt output.

NOTE: Only one of the PCI interrupt outputs, LINT[0], has the drive strength to be fully compliant with the PCI specification. The other seven may require buffering if they are to be routed to PCI compliant interrupt lines. For most applications, however, the drive strength provided should be sufficient.

PCI interrupts may be generated from multiple sources:

• VME interrupts
-IRQ*[7:1]
-SYSFAIL* (SYSFAIL bit)
-ACFAIL*

• internal interrupts
-DMA
-VME bus error encountered
-PCI bus error encountered
-VMEbus ownership has been granted while VOWN bit is set
-software interrupt
-VME IACK cycle performed in response to a software interrupt

Each of these sources may be individually enabled in the LINT_EN register (Universe II Register Definitions, Table A.40) and mapped to a single LINT signal through the LINT_MAP0 and LINT_MAP1 registers (Universe II Register Definitions, Table A.42 and Table A.43). When an interrupt is received on any of the enabled sources, the Universe II asserts the appropriate LINT pin and sets a matching bit in the LINT_STAT register (Universe II Register Definitions, Table A.41). See Table 10-1 below for a list of the enable, mapping and status bits for PCI interrupt sources.



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

| Interrupt Source | Enable Bit in LINT_EN | Mapping Bit in LINT_MAP0 or LINT_MAP1 | Status Bit in LINT_STAT |
|---|---|---|---|
| ACFAIL* | ACFAIL | ACFAIL | ACFAIL |
| SYSFAIL* | SYSFAIL | SYSFAIL | SYSFAIL |
| PCI Software Interrupt | SW_INT | SW_INT | SW_INT |
| VME Software IACK | SW_IACK | SW_IACK | SW_IACK |
| VMEbus Error | VERR | VERR | VERR |
| PCI Bus Error | LERR | LERR | LERR |
| DMA Event | DMA | DMA | DMA |
| VMEbus Interrupt Input | VIRQ7-1 | VIRQ7-1 | VIRQ7-1 |
| VMEbus Ownership | VOWN | /VOWN | VOWN |

Table 10-1 PCI Interrupt Source Enabling, Mapping, and Status Bits

The LINT_STAT register shows the status of all sources of PCI interrupts, independent of whether that source has been enabled. This implies that an interrupt handling routine must mask out those bits in the register that do not correspond to enabled sources on the active LINT pin.

Except for SYSFAIL* and ACFAIL*, all sources of PCI interrupts are edge sensitive. Enabling of the ACFAIL* or SYSFAIL* sources (ACFAIL and SYSFAIL bits in the LINT_EN register) cause the status bit and mapped PCI interrupt pin to assert synchronously with the assertion of the ACFAIL* or SYSFAIL* source. The PCI interrupt is negated once the ACFAIL* or SYSFAIL* source is also negated and its respective status bit in the LINT_STAT register is cleared, or the interrupt is disabled. Both of these sources are synchronized and filtered by the 64 MHz clock at their inputs.

All other sources of PCI interrupts are edge sensitive. Note that the VMEbus source for PCI interrupts actually comes out of the VMEbus Interrupt Handler block and reflects acquisition of a VME STATUS/ID. Therefore, even though VMEbus interrupts externally are level sensitive as required by the VME specification, they are internally mapped to edge-sensitive interrupts (see “VMEbus Interrupt Handling” on page 10-7).

The interrupt source status bit (in the LINT_STAT register) and the mapped LINT pin remain asserted with all interrupts. The status bit and the PCI interrupt output pin are only released when the interrupt is cleared by writing a “one” to the appropriate status bit.



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

## VMEbus Interrupt Generation

Interrupts may be generated on any combination of VME interrupts from multiple sources:

• PCI Interrupts
- LINT[7:0]

• Internal Interrupts
- DMA
- VME bus error encountered
- PCI bus error encountered
- software interrupt

Each of these sources may be individually enabled through the VINT_EN register (Universe II Register Definitions, Table A.44) and mapped to a particular VME Interrupt level using the VINT_MAP0 and VINT_MAP1 registers (Universe II Register Definitions, Table A.46 and Table A.47). Multiple sources may be mapped to any VME level. Mapping interrupt sources to level 0 effectively disables the interrupt.

Once an interrupt has been received from any of the sources, the Universe II sets the corresponding status bit in the VINT_STAT register (Universe II Register Definitions, Table A.45), and asserts the appropriate VMEbus interrupt output signal (if enabled). When a VME interrupt handler receives the interrupt, it will perform an IACK cycle at that level. When the Universe II decodes that IACK cycle together with IACKIN* asserted, it provides the STATUS/ID previously stored in the STATID register (Universe II Register Definitions, Table A.48). See Table 10-2 below for a list of the enable, mapping and status bits for VMEbus interrupt sources.

| Interrupt Source | Enable Bit in VINT_EN | Mapping Bit in VINT_MAP0 or VINT_MAP1 | Status Bit in VINT_STAT |
|---|---|---|---|
| VME Software Interrupt | SW_INT | SW_INT | SW_INT |
| VMEbus Error | VERR | VERR | VERR |
| PCI Bus Error | LERR | LERR | LERR |
| DMA Event | DMA | DMA | DMA |
| PCI bus Interrupt Input | LINT7-0 | LINT7-0 | LINT7-0 |

Table 10-2 VMEbus Interrupt Source Enabling, Mapping, and Status Bits



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

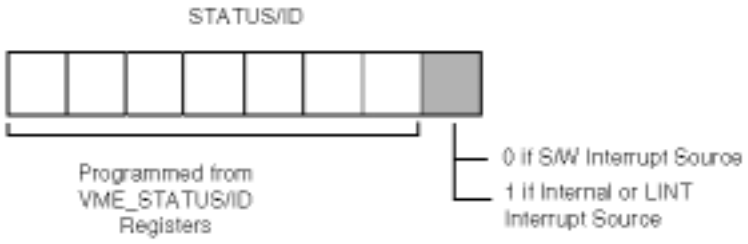



Figure 10-2 Status/ID Provided by the Universe II

For all VMEbus interrupts, the Universe II interrupter supplies a pre-programmed 8-bit STATUS/ID: a common value for all interrupt levels. The upper seven bits are programmed in the STATID register. The lowest bit is cleared if the source of the interrupt was the software interrupt, and is set for all other interrupt sources. If a software interrupt source and another interrupt source are active and mapped to the same VMEbus interrupt level, the Universe II gives priority to the software source.

Once the Universe II has provided the STATUS/ID to an interrupt handler during a software initiated VMEbus interrupt, it generates an internal interrupt, SW_IACK. If enabled, this interrupt feeds back to the PCI bus (through one of the LINT pins) to signal a process that the interrupt started through software has been completed.

All VMEbus interrupts generated by the Universe II are RORA, except for the software interrupt which is ROAK. This means that if the interrupt source was the software interrupt, then the VMEbus interrupt output is automatically negated when the Universe II receives the IACK cycle. However, for any other interrupt, the VMEbus interrupt output remains asserted until cleared by a register access. Writing a “one” to the relevant bit in the VINT_STAT register clears that interrupt source. However, since PCI interrupts are level sensitive, if an attempt is made to clear the VMEbus interrupt while the LINT pin is still asserted, the VMEbus interrupt remains asserted. This causes a second interrupt to be generated to the VMEbus. For this reason, a VME interrupt handler should clear the source of the PCI interrupt before clearing the VMEbus interrupt.

## PCI Interrupt Handling

All eight PCI interrupt lines, LINT[7:0], can act as interrupt inputs to the Universe II. They are level sensitive and, if enabled in the VINT_EN register (Table A.44), immediately generate an interrupt to the VMEbus. It is expected that when a VMEbus interrupt handler receives the Universe II’s STATUS/ID from the Universe II, the interrupt handler will clear the VMEbus interrupt by first clearing the source of the interrupt on the PCI bus, and then clearing the VMEbus interrupt itself (by writing a “one” to the appropriate bit in the VINT_STAT register, Table A.45).

Note that since PCI interrupts are level sensitive, if an attempt is made to clear the VMEbus interrupt while the LINT pin is still asserted, the VMEbus interrupt



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

remains asserted. This causes a second interrupt to be generated to the VMEbus. For this reason, a VMEbus interrupt handler should clear the source of the PCI interrupt before clearing the VMEbus interrupt.

## VMEbus Interrupt Handling

As a VME interrupt handler, the Universe II can monitor any or all of the VMEbus interrupt levels. It can also monitor SYSFAIL* and ACFAIL*, although IACK cycles are not generated for these inputs. Each interrupt is enabled through the LINT_EN register (Universe II Register Definitions, Table A.40).

Once enabled, assertion of any of the VMEbus interrupt levels, IRQ[7:1]*, causes the internal interrupt handler circuitry to request ownership of the Universe II's VME Master Interface on the level programmed in the MAST_CTL register . This interface is shared between several channels in the Universe II: the PCI Slave Channel, the DMA Channel, and the Interrupt Channel. The Interrupt Channel has the highest priority over all other channels and, if an interrupt is pending, assumes ownership of the VME Master Interface when the previous owner has relinquished ownership.

There may be some latency between reception of a VMEbus interrupt and generation of the IACK cycle. This arises because of the latency involved in the Interrupt Channel gaining control of the VME master interface, and because of possible latency in gaining ownership of the VMEbus if the VME Master Interface is programmed for release-when-done. In addition, the Universe II only generates an interrupt on the PCI bus once the IACK cycle has completed on the VMEbus. Because of these combined latencies (time to acquire VMEbus and time to run the IACK cycle), systems should be designed to accommodate a certain worst case lag time from VMEbus interrupt generation to its translation to the PCI bus.

When the Universe II receives a STATUS/ID in response to an IACK cycle, it stores that value in one of seven registers. These registers, V1_STATID through V7_STATID (Universe II Register Definitions, Table A.49 to Table A.55), store the STATUS/ID corresponding to each IACK level (in the STATID field). Once an IACK cycle has been generated and the resulting STATUS/ID is latched, another IACK cycle will not be run on that level until the level has been re-armed by writing a “one” to the corresponding status bit in the VINT_STAT register (Universe II Register Definitions, Table A.45). If other interrupts (at different levels) are pending while the interrupt is waiting to be re-armed, IACK cycles are run on those levels in order of priority and the STATUS/IDs stored in their respective registers.



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

Once the IACK cycle is complete and the STATUS/ID stored, an interrupt is generated to the PCI bus on one of LINT#[7:0] interrupt output lines depending on the mapping for that VME level in the LINT_MAP0 register. The interrupt is cleared and the VMEbus interrupt level is re-armed by clearing the correct bit in the LINT_STAT register.

## Bus Error During IACK

A bus error encountered on the VMEbus while the Universe II is performing an IACK cycle is handled by the Universe II in two ways. The first is through the error logs in the VME Master Interface. These logs store address and command information whenever the Universe II encounters a bus error on the VMEbus . If the error occurs during an IACK cycle, the IACK# bit is set in the V_AMERR register (Universe II Register Definitions, Table A.80). The VME Master Interface also generates an internal interrupt to the Interrupt Channel indicating a VMEbus error occurred. This internal interrupt can be enabled and mapped to either the VMEbus or PCI bus.

As well as generating an interrupt indicating an error during the IACK cycle, the Universe II also generates an interrupt as though the IACK cycle completed successfully. If an error occurs during the fetching of the STATUS/ID, the Universe II sets the ERR bit in the Vn_STATID register (Universe II Register Definitions, Table A.49 to Table A.55), and generates an interrupt on the appropriate LINT pin (as mapped in the LINT_MAP0 register, Universe II Register Definitions, Table A.42). The PCI resource, upon receiving the PCI interrupt, is expected to read the STATUS/ID register, and take appropriate actions if the ERR bit is set. Note that the STATUS/ID cannot be considered valid if the ERR bit is set in the STATUS/ID register.

It is important to recognize that the IACK cycle error may generate two PCI interrupts: one through the VMEbus master bus error interrupt and another through the standard PCI interrupt translation. Should an error occur during acquisition of a STATUS/ID, the VINT_STAT register (Universe II Register Definitions, Table A.45) will show that both VIRQx, and VERR are active.

## Internal Interrupt Handling

The Universe II's internal interrupts are routed from several processes in the device (see Figure 10-3 on page 10-9). There is an interrupt from the VME master interface to indicate a VMEbus error, another from the PCI master interface to indicate an error on that bus, another from the DMA to indicate various conditions in that channel, along with several others as indicated in Table 10-3 below. Some interrupts may be routed only to the PCI bus, some only to the VMEbus, and others that may be routed to both (although we recommend that you map interrupts to a single bus).



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

| Interrupt Source | May be Routed to: | |
|---|---|---|
| | VMEbus | PCI Bus |
| PCI bus errors | √ | √ |
| VME bus error | √ | √ |
| DMA event | √ | √ |
| IACK cycle complete for s/w interrupt | | √ |
| VME bus ownership granted | | √ |
| PCI s/w interrupt | | √ |
| VME s/w interrupt | √ | |

Table 10-3 Internal Interrupt Routing

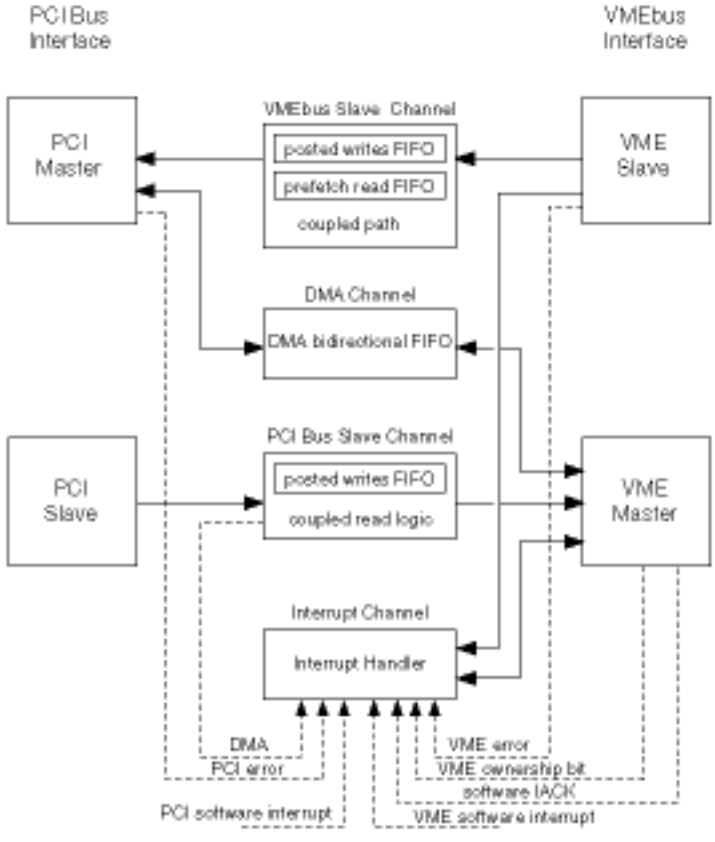



Figure 10-3 Sources of Internal Interrupts



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

## VME and PCI Software Interrupts

Two interrupt sources are provided as mechanisms to interrupt the PCI or VME buses through software. These interrupts may be triggered by writing a “one” to the respective enable bits. Writing a “one” to the SW_INT bit in the VINT_EN register (Universe II Register Definitions, Table A.44) triggers one (and only one) interrupt on the VMEbus on the level programmed in the VINT_MAP1 register (Universe II Register Definitions, Table A.47). When an IACK cycle is serviced on the VMEbus, the Universe II can be programmed to generate an interrupt on the PCI bus by setting the SW_IACK enable bit in the LINT_EN register. Writing a “one” to the SW_INT in the LINT_EN (Universe II Register Definitions, Table A.40) registersimilarly generates an interrupt to the PCI bus. While these interrupt sources are active, their respective status bits are set (SW_INT in VINT_STAT; SW_IACK and SW_INT in LINT_STAT). Either interrupt may be cleared by clearing the respective enable bit although clearing a VMEbus interrupt through this method is not recommended since it may result in a spurious interrupt on that bus.

Since the software interrupt is edge-sensitive, the SW_INT bit should be cleared before generating another interrupt. It is recommended that the appropriate interrupt handler clear this bit once it has completed its operations. Alternatively, the process generating the interrupt could clear this bit before re-asserting it.

Software interrupts on the VMEbus have priority over other interrupts mapped internally to the same level on the VMEbus. When a VMEbus interrupt handler generates an IACK cycle on a level mapped to both a software interrupt and another interrupt, the Universe II always provides the STATUS/ID for the software interrupt (bit zero of the Status/ID is cleared). If there are no other active interrupts on that level, the interrupt is automatically cleared upon completion of the IACK cycle (since software interrupts are ROAK).

While the software interrupt STATUS/ID has priority over other interrupt sources, the user can give other interrupt sources priority over the software interrupt. This is done by reading the LINT_STAT register (Universe II Register Definitions, Table A.41) when handling a Universe II interrupt. This register indicates all active interrupt sources. Using this information, the interrupt handler can then handle the interrupt sources in any system-defined order.



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

## Software IACK Interrupt

The Universe II generates an internal interrupt when it provides the software STATUS/ID to the VMEbus. This interrupt can only be routed to a PCI interrupt output. A PCI interrupt will be generated upon completion of an IACK cycle that had been initiated by the Universe II's software interrupt if the SW_IACK bit in the LINT_EN register (Universe II Register Definitions, Table A.40) is set and the SW_IACK field in the LINT_MAP1 register (Universe II Register Definitions, Table A.43) is mapped to a corresponding PCI interrupt line. This interrupt could be used by a PCI process to indicate that the software interrupt generated to the VMEbus has been received by the slave device and acknowledged.

Like other interrupt sources, this interrupt source can be independently enabled through the LINT_EN register (Universe II Register Definitions, Table A.40) and mapped to a particular LINT pin using the LINT_MAP1 register (Universe II Register Definitions, Table A.43). A status bit in the LINT_STAT register (Universe II Register Definitions, Table A.41) indicates when the interrupt source is active, and is used to clear the interrupt once it has been serviced.

## VME Ownership Interrupt

The VMEbus ownership interrupt is generated when the Universe II acquires the VMEbus in response to programming of the VOWN bit in the MAST_CTL register (Universe II Register Definitions, Table A.56). This interrupt source can be used to indicate that ownership of the VMEbus is ensured during an exclusive access. The interrupt is cleared by writing a one to the matching bit in the LINT_STAT register (Universe II Register Definitions, Table A.41).



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

*This page is intentionally blank.*



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

# CHAPTER 11 Specifications

**General**

| | |
|---|---|
| *Model* | V5B |
| *Description* | PC/AT Compatible VMEbus Single Board Computer |
| *Hardware Compatibility* | VMEbus Dual Eurocard, VME64 ANSI/VITA 1-1994 |

**VMEbus**

| | |
|---|---|
| *Controller* | Tundra Universe II |
| *Configuration* | DTB Master, Option A32/A24/A16, D32/D16/D08(EO), RMW<br>DTB Slave, Option A32/A24/A16, D32/D16/D08(EO), RMW |
| *Interrupter* | Programmable, 1-of-7 |
| *Interrupt Handler* | Programmable, IH(1-7) |
| *Requester* | Programmable, BR(3,2,1,0), Option ROR and RWD |
| *Arbiter* | RRS, PRI, SGL |
| *Block Mode Transfer* | Master/Slave BLT and MBLT D64/D32/D16 |

**PCI Bus**

| | |
|---|---|
| *Controller* | ALI Aladdin III Chip Set |
| *Clock Rate* | 33MHz |

**PC/AT Bus**

| | |
|---|---|
| *Controller* | ALI Aladdin III Chip Set |
| *Clock Rate* | 7.16 MHz |

**Serial Interface**

| | |
|---|---|
| *Controller* | SMC FDC37C665 |
| *Number* | 2 Com Ports |
| *Compatibility* | IBM PC |
| *Interface Type* | COM1: RS-232; COM2: RS-232/422/485 |
| *Signals Provided* | RxD, TxD, RTS, CTS, DSR, DTR, DCD, RI |
| *Connector* | Micro-Miniature DB9P |

**Parallel Interface**

| | |
|---|---|
| *Controller* | SMC FDC37C665 |
| *Number* | 1 Parallelo Port |
| *Compatibility* | Centronics Parallel, PS/2 Bidirectional |
| *Interface* | Buffered Parallel |
| *Connector* | Micro-Miniature DB25S |

**Mouse Interface**

| | |
|---|---|
| *Controller* | AMI 8742 |
| *Compatibility* | Microsoft Mouse |
| *Connector* | 6-pin Circular Mini-DIN |

**Keyboard Interface**

| | |
|---|---|
| *Controller* | AMI 8742 |
| *Compatibility* | IBM PC/AT |
| *Connector* | 6-pin Circular Mini-DIN (PS/2 Style) Standard PC/AT Cable Provided |



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

**Video Interface**

| | |
|---|---|
| *Controller* | Cirrus CL-GD5436 |
| *Compatibility* | VGA/SVGA |
| *Capability* | 1280×1024×256 colors, 1024x768x64k, 800×600×16.7M, 680x480x16.7M colors |
| *Display Supported* | Color CRT |
| *Connector* | HD15S |
| *Video Memory* | 2 Megabytes of High Speed VRAM |

**Disk Drive Interface**

| | |
|---|---|
| *Floppy Disk* | ST412 Interface, SMC FDC37C665 Controller, Provided Through P2 |
| *Hard Disk* | Bus Mastering IDE Interface Provided Through P2<br>Fast SCSI-2 Interface Provided Through P2 |

**Ethernet Interface**

| | |
|---|---|
| *Type* | IEEE 802.3 10Base2 (Thinwire), Provided Through Front Panel BNC<br>Optional 10/100BaseT (Twisted-Pair), Provided Through Front Panel RJ45 |

**Electrical**

| | |
|---|---|
| *Power* | +5VDC @ 5–10A (memory & CPU dependent)<br>+12VDC @ 200mA (maximum)<br>-12VDC @ 100mA (maximum) |

**Physical**

| | |
|---|---|
| *Size* | 160mm × 233mm (Dual Eurocard), 6U×4HP |
| *Construction* | Multi-Layer Printed Circuit, 8 Layers |

**Environmental**

| | |
|---|---|
| *Temperature* | 0 to 55° Celsius Inlet Air, Operating<br>-40 to 85° Celsius, Non-Operating |
| *Cooling* | Forced Air, 100LFM Fan Recommended |
| *Humidity* | 10 to 95% Relative Humidity, Non-Condensing |
| *Shock* | 6G Max. Operating, 10G Max. Non-Operating |



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

# CHAPTER 12 Support, Service, and Warranty

## Chapter Scope

The following sections describe SBS Embedded Computers product support program. It states our product warranty terms and provides details about what action to take if you experience a problem with the product.

## Warranty Statement

SBS Embedded Computers VMEbus products come with a “return-to-factory” warranty which covers defects in materials and workmanship for a period of two years from the date of product shipment to the customer (original purchaser), provided the product is unmodified and has been subject to normal and proper use. This warranty applies to all standard board-level products which do not incorporate disk drives. Products which incorporate floppy or hard disk drives are also warranted for two years with the exception of the drives themselves. The drives will be warranted for a period of ninety days, as is the normal period for electromechanical components. SBS Embedded Computers makes no warranty or representation, express or implied, with respect to software, its performance, quality, or fitness for a particular purpose. This does not include the media on which the software is distributed, which carries a warranty covering defects in materials and workmanship for a period of ninety days.

## If You Have a Problem with an SBS Product

Free technical support is available by phone, fax or email. Telephone suppport is available during the following Eastern Time hours:

Monday through Friday 8:30 am - 5:30 pm.

You can reach technical support at (919) 851-1101 voice, (919) 851-2844 fax or email at support@sbs-ec.com.



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

## Product Repairs

To expedite assistance for problems, be able to provide the following:

- Your Name, Phone and Company.
- Product with which you are having trouble.
- Serial Number and Revision. (located on the P2 Connector)
- Operating system you are running.
- Detailed description of your problem and any error messages that have appeared on the screen.

Depending on the circumstances of the problem, it may be deemed necessary to return the product to SBS Embedded Computers for repair. In order to return the product for repair, the following steps are necessary:

1. Obtain a Return Material Authorization number (RMA#) from SBS Customer Support.

2. Ship the product prepaid to the designated repair point.

3. Provide a written description of the claimed defect with the product.

### Obtaining an RMA Number

To obtain a product return authorization number (RMA#), you should call our Customer Service department through our main number.

### Shipping the Product

Any product returned to SBS should be in its original shipping carton if possible. Otherwise the product should be carefully packaged in a conductive packing material and placed in a cushioned corrugated carton suitable for shipping. Please mark the shipping label with the RMA number and return it to:

Customer Service Department
Att: RMA# *(put RMA number here)*
SBS Embedded Computers, Inc.
6301 Chapel Hill Road
Raleigh, NC 27607



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

### Providing a Product Defect Report

When you are returning a product for repair, it is very important to include a written report which details the nature of the problem in order to expedite the repair. Please make sure that the following information is included:

- RMA Number
- Product:
- Serial Number
- Contact:
- Phone
- Description of the Problem/Defect

### Warranty Repairs

Any product returned and found to be under warranty will be repaired or replaced at the discretion of SBS within two working days of receipt and shipped freight prepaid to the Customer.

### Non-Warranty Repairs

If a product is found not to be under warranty, we will notify you of the non-warranty situation and provide you with a fixed cost and a schedule for the repair. Non-warranty repairs generally require that a purchase order be issued to SBS Embedded Computers Inc. for the amount of the repair before repairs are undertaken.




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

*This page is intentionally blank.*

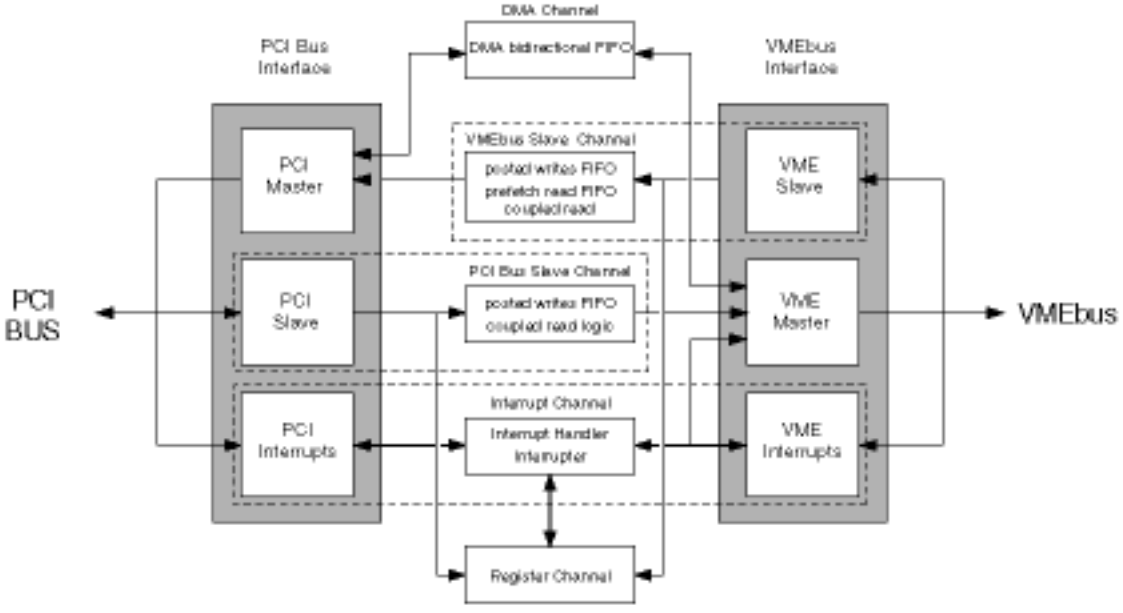


Artisan Technology Group - Quality Instrumentation ... Guaranteed | (888) 88-SOURCE | www.artisantg.com

# APPENDIX A Universe II Block Diagram

## Appendix Scope

This appendix gives a functional block diagram of the Universe II VME Interface Controller. The relationships between PCI Master and PCI Slave Modules, VMEbus Master and VMEbus Slave Modules, and VME/PCI Slave Channels are shown. Also, the Interrupt Channel and Register Channel are illustrated.






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

*This page is intentionally blank.*



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

# APPENDIX B VME BIOS Routines

This section is not applicable to the Award BIOs

---

## Appendix Scope

This appendix describes in detail, on a function by function basis, the routines which comprise the V5B VME BIOS. The following VME BIOS functions are accessible from software interrupt 69H:

Note: all function numbers are given in hexadecimal

```
**********************************************************************
function 00h - GET_VME_BIOS_VERSION
Returns the VME BIOS version, board type and A32 capability

ENTRY: AH = 00H
EXIT:  AL = VERSION OF THIS FIRMWARE, VERSION & MAJOR REV, IN BCD format
       AH = VME SYSTEM/OPTION INFO.:

              bit:         if set:                if clear:
              7-5          Reserved
              4-1          0000 = XT
                           0001 = XP
                           0010 = XPS
                           0011 = XPX
                           0100 = VME-1486
                           0101 = VME-7486
                           0110 = VME-2486
                           0111 = VME-8486
                           1000 = VME-3486
                           1001 = VME-9586
                           1010 = V5A
                           1110 = V5B
                           1011 = VME-190
                           1100 = IV-586
              0            EXTENDED ADDRESS       NO EXTENDED
                           CAPABLE                ADDRESS CAPABILITY
```



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

```
************************************************************************
function 01h - REQUEST_VME_INTERRUPT
Function generates a VME IRQ to the VMEbus.

ENTRY: AH = 01H
       AL = VMEBUS INTERRUPT REQUEST LEVEL (1-7)
       BL = VMEBUS INTERRUPT VECTOR NUMBER (ID/Status Byte)
EXIT:  NOTHING


************************************************************************
function 02h - GET_VME_INTERRUPT_STATUS
Function returns the pending VME IRQ interrupt status.

ENTRY: AH = 02H
EXIT:  AL = VMEBUS INTERRUPT LEVEL PENDING, 0 returned if none pending,

************************************************************************
function 03h - SETUP_VME_INTERRUPTS
Function enables VME IRQs to pass through to the PCI (local) bus.

ENTRY: AH = 03H
       AL = any bit 1-7 set enables the corresponding VME IRQ
       BL = for each bit set in AL, the corresponding bit in BL tells
            which PCI (local) interrupt path to take.  0 means LINT0 (PCI
            INTA), 1 means LINT1 (PCI Steerable interrupt 0).
EXIT:  NOTHING

************************************************************************
function 0Bh - CLEAR VME INTERRUPTS
Function disables all VME interrupts

ENTRY: AH = 0BH
EXIT:  NOTHING

************************************************************************
function 0Ch - GET_SYSFAIL_STATUS
Function returns the status of the VMEbus SYSFAIL signal

ENTRY: AH = 0CH
EXIT:  AL BIT 0 = 1 IF SYSFAIL IS ASSERTED
       AL BIT 0 = 0 IF SYSFAIL IS DEASSERTED

************************************************************************
function 0Dh - SET_SYSFAIL_STATUS
Function will set or clear the SYSFAIL signal

ENTRY: AH = 0DH
       AL BIT 0 = 1 TO ASSERT SYSFAIL
       AL BIT 0 = 0 TO DEASSERT SYSFAIL
EXIT:  NOTHING
```



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

```
*********************************************************************
function 10h - SERVICE_INTERRUPTS
Function processes the pending VMEbus interrupt.

ENTRY: AH = 10H
EXIT:  AL:
           Bit  Meaning if set
            0   highest priority vme interrupt was serviced
            1   SYSFAIL Interrupt occurred
            2   ACFAIL Interrupt occurred
            3   DMA error occurred
            4   PCI error occurred
            5   VME error occurred
            6   VOWN Interrupt occurred
            7   Reserved
       BL:
           bits 0-6 = VME IRQ (1-7) acknowleged
            bit 7    = BUS error occurred after int vector was read
       CL: VME Interrupt Status/ID byte;  0 if bit-5 in AL is set
       EOI's acknowleged.

*********************************************************************
function 12h - GET_MASTER_IMAGE
Function will return information about a VME Master Image.

ENTRY: AH = 12H
       AL = image number (0-3)

EXIT:  AL = control status:
             bit 7 = image enabled
             bit 6 = Posted Write enabled
             bits 5-3 = VME address space (000 = A16, 001 = A24,
                            010 = A32, 110 = user1, 111 = user2)
             bit 2 = Data/PGM   (0 = data, 1 = program)
             bit 1  = Privilege (0 = user, 1 = supervisory)
             bit 0  = Block Transfer enabled
       BX = A31-A16 of Base address
       CX = A31-A16 of Bound address
       DX = A31-A16 of translation offset
       AH = A15-A12 of Base address in bits 7-4, A15-A12 of Bound
                      address in bits 3-0 (image 0 only)
       DI = A15-A12 of Translation offset in bits 3-0 (image 0 only)

*********************************************************************
function 13h - PUT_MASTER_IMAGE
Function will setup parameters for a VME Master Image

ENTRY: AH = 13H
       AL = image number (0-3)

       DI = control status:
              bit 7 = enable image
              bit 6 = Posted Write enabled
              bits 5-3 = VME address space (000 = A16, 001 = A24,
                             010 = A32, 110 = user1, 111 = user2)
              bit 2 = Data/PGM   (0 = data, 1 = program)
              bit 1  = Privilege (0 = user, 1 = supervisory)
              bit 0  = Block Transfer enabled
       SI (image 0 only) = odd bytes of addresses
              A15-A12 of Base address in bits 11-8, A15-A12 of Boundary
              address in bits 7-4, A15-A12 of Translation offset in bits 3-0
       BX = A31-A16 of Base address
       CX = A31-A16 of Bound address
       DX = A31-A16 of translation offset
EXIT:  NOTHING
```



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

```
**********************************************************************
function 14h - GET_SLAVE_IMAGE
Function will return information about a VME Slave Image

ENTRY: AH = 14H
       AL = Image number (0-3)
EXIT:  DI = control status:
             bit 10 = image enabled
             bit  9 = Posted Write enabled
             bit  8 = Prefetch Read enabled
             bits 7-5 = VME address space (000 = A16, 001 = A24,
                        010 = A32, 110 = user1, 111 = user2)
             bits 4&3 = Data/PGM (01 = data, 10 = program, 11 = both)
             bits 2&1 = Privilege (01 = user, 10 = supervisory, 11 = both)
             bit 0  = PCI Lock on VME RMW enabled
       BX = A31-A16 of Base address
       CX = A31-A16 of Bound address
       DX = A31-A16 of translation offset
       AH = A15-A12 of Base address in bits 7-4, A15-A12 of Bound
                    address in bits 3-0 (image 0 only)
       AL = A15-A12 of Translation offset in bits 3-0 (image 0 only)


**********************************************************************
function 15h - PUT_SLAVE_IMAGE
Function will setup parameters for a VME Slave Image

ENTRY: AH = 15H
       AL = Image number (0-3)
       DI = control status:
             bit 10 = enable image
             bit  9 = Posted Write enabled
             bit  8 = Prefetch Read enabled
             bits 7-5 = VME address space (000 = A16, 001 = A24,
                        010 = A32, 110 = user1, 111 = user2)
             bits 4&3 = Data/PGM (01 = data, 10 = program, 11 = both)
             bits 2&1 = Privilege (01 = user, 10 = supervisory, 11 = both)
             bit 0  = Enable PCI Lock on VME RMW
       BX = A31-A16 of Base address
       CX = A31-A16 of Bound address
       DX = A31-A16 of translation offset
       SI = A15-A12 of Base address in bits 7-4, A15-A12 of Bound
                    address in bits 3-0, A15-A12 of Translation
                    offset in bits 11-8 (image 0 only)
EXIT:  NOTHING

**********************************************************************
function 16h - GET_EEPROM_WORD
Function returns a word from the on-board serial EEPROM.

ENTRY: AH = 16H
       AL = Word offset into the serial EEPROM (0-63)
EXIT:  AX = EEPROM data from the requested OFFSET

**********************************************************************
function 17h - SET_EEPROM_WORD
Function writes a word to the on-board serial EEPROM.
WARNING: This device is used internally to store board information.
         Writing to the wrong location will interfere with proper board
         performance!

ENTRY: AH = 17H
       AL = Word offset into the serial EEPROM (0-63)
       BX = Data to write to the device
EXIT:  NOTHING
```



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

```
************************************************************************
function 18h - GET_SERIAL_SWITCH
Function returns the 8 on-board serial switch values.

ENTRY: AH = 18H
EXIT:  AL = Bits 0-7 correspond to serial switches 1-8 respectively.

************************************************************************
function 19h - SET_SERIAL_SWITCH
Function writes to the on-board serial switches.
WARNING: These switches are used to run certain on-board resources.
         Writing the wrong value to these switches will render the board
         inoperable.

ENTRY: AH = 19H
       AL = Bits 0-7 correspond to switches 1-8 respectively
EXIT:  NOTHING

************************************************************************
function 1fh - SET_BOARD
Function loads all VME parameters from saved (EEPROM) values.

ENTRY: AH = 1FH
EXIT:  NOTHING

************************************************************************
function 20h - Read VMEbus.
Read the VMEbus in real mode (up to 64Kb)

ENTRY: AH = 20H
       AL = transfer parameters:
            bit 4 = mode: 0=non-privileged  1=supervisor
            bit 3 = pgm/data: 0=data  1=program
            bits 2-0 = VME address space: 000=A16  001=A24  010=A32
                                          110=User1  111=User2
       BX = VMEbus address bits A31-A16
       SI = VMEbus address bits A15-A0
       CX = number of bytes to transfer
       DS:DI = PCI bus (local) address

EXIT:     Nothing.

************************************************************************
function 21h - Write VMEbus.
Write to the VMEbus in real mode (up to 64Kb)

ENTRY: AH = 21H
       AL = transfer parameters:
            bit 4 = mode: 0=non-privileged  1=supervisor
            bit 3 = pgm/data: 0=data  1=program
            bits 2-0 = VME address space: 000=A16  001=A24  010=A32
                                          110=User1  111=User2
       BX = VMEbus address bits A31-A16
       DI = VMEbus address bits A15-A0
       CX = number of bytes to transfer
       DS:SI = PCI bus (local) address

EXIT:     Nothing.
```



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

```
**********************************************************************
function 24h - Enable Universe II Registers to the VMEbus.
Function will enable the UNIVERSE registers to appear on the VMEbus

ENTRY: AH = 24H
       AL = bit 7 = enable if set, disable if clear
            bits 6&5 = pgm/data: 01=data, 10=program, 11=both
            bits 4&3 = mode: 01=non-privileged  10=privileged, 11=both
            bits 2-0 = VME address space: 000=A16  001=A24  010=A32
       BX = VMEbus address bits A31-A16
       CL = VMEbus address bits A15-A12 in CL bits 7-4

EXIT:     Nothing.
```



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

# APPENDIX C V5B Register Summary

## Appendix Scope

This appendix gives a summary of the V5B board-specific registers which are not standard PC/AT, or peripheral device registers. The register layout and location are given, as are the bit mneumonics and names. Read or write capability for each bit is also shown.

| RES | RES | RES | FPSR | RES | RES | RES | IGSR |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

## Reset Control Register I/O 2A0H

| Bit | Description |
|---|---|
| 0 | Ignore SYSRESET from VME (R/W) |
| 4 | Front Panel Reset to VME SYSRESET Enable (R/W)**Bit** |



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

## BIOS Control Register I/O 2A1H

| BIOS | SBWE | OBWE | RES | RES | RES | RSSL | SCSI |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

**Description**

5 Socketed BIOS Write Enable (R/W)

6 Onboard BIOS Write Enable (R/W)

## User Register I/O 2A2H

| SYS | Reserved | | | | | | |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Bit | Description |
|---|---|
| 7 | SYS LED Control (R/W) |

## Serial EEPROM Register I/O 2A3H

| RES | EECS | EEDIN | RES | RES | RES | SD0 | CLK |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| Bit | Description |
|---|---|
| 0 | Serial Clock (R/W) |
| 1 | Serial Data Out (R/W) |
| 2 | Reserved |
| 3 | Reserved |
| 4 | Reserved |
| 5 | EEPROM Data In (R/W) |
| 6 | EEPROM Chip Select (R/W) |

## Byte Swapping I/O 2A4H

| RES | RES | RES | STAT | BYM1 | BYM0 | BYS1 | BYS0 |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

| | Master | | Slave | |
|---|---|---|---|---|
| | **BYM1** | **BYM0** | **BYS1** | **BYS0** |
| M32 | 0 | 1 | 0 | 1 |
| I32 | 0 | 0 | 0 | 0 |
| I16 | 1 | 0 | 1 | 0 |

| Bit | Description |
|---|---|
| 0-1 | These bits control the byte swap mode when doing a VME Slave transfer. |
| 2-3 | These bits control the byte swap mode when doing a VME Master transfer. |
| 4 | If bits 0-3 are modified, the new mode is not in effect until the VMEbus is inactive. Bit 4 of the byte swapping register is read-only and indicates when the mode specified by by bits 0-3 is in effect. |



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

# APPENDIX D VME Address Modifier Codes

The following table shows the standard VME address modifier codes. Codes 10H through 1FH are user-defined. All other codes are reserved.

| Address Modifier | Address Size | Transfer Type |
|---|---|---|
| 3FH | 24 | Standard Supervisory Block Transfer (BLT) |
| 3EH | 24 | Standard Supervisory Program Access |
| 3DH | 24 | Standard Supervisory Data Access |
| 3BH | 24 | Standard Non-Privileged Block Transfer (BLT) |
| 3AH | 24 | Standard Non-Privileged Program Access |
| 39H | 24 | Standard Non-Privileged Data Access |
| 38H | 24 | Standard Non-Privileged 64-bit Block Transfer (MBLT) |
| 37H | 40 | A40 BLT |
| 35H | 40 | A40 Lock Command (LCK) |
| 34H | 40 | A40 Access |
| 32H | 24 | A24 Lock Command (LCK) |
| 2FH | N/A | Configuration ROM/Control & Status Register |
| 2DH | 16 | Short Supervisory Access |
| 2CH | 16 | A24 Lock Command (LCK) |
| 29H | 16 | Short Non-Privileged Access |
| 0FH | 32 | Extended Supervisory Block Transfer (BLT) |
| 0EH | 32 | Extended Supervisory Program Access |
| 0DH | 32 | Extended Supervisory Data Access |
| 0CH | 32 | Extended Supervisory 64-bit Block Transfer (MBLT) |
| 0BH | 32 | Extended Non-Privileged Block Transfer (BLT) |
| 0AH | 32 | Extended Non-Privileged Program Access |
| 09H | 32 | Extended Non-Privileged Data Access |
| 08H | 32 | Extended Non-Privileged 64-bit Block Transfer (MBLT) |
| 05H | 32 | A32 Lock Command (LCK) |
| 04H | 64 | A64 Lock Command (LCK) |
| 03H | 64 | A64 Block Transfer (BLT) |
| 01H | 64 | A64 Single Transfer Access |
| 00H | 64 | A64 64-bit Block Transfer (MBLT) |

Table E-1 Address Modifier Codes



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

*This page is intentionally blank.*



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

Artisan Technology Group

# Artisan Technology Group is your source for quality new and certified-used/pre-owned equipment

- FAST SHIPPING AND DELIVERY
- TENS OF THOUSANDS OF IN-STOCK ITEMS
- EQUIPMENT DEMOS
- HUNDREDS OF MANUFACTURERS SUPPORTED
- LEASING/MONTHLY RENTALS
- ITAR CERTIFIED SECURE ASSET SOLUTIONS

**SERVICE CENTER REPAIRS**
Experienced engineers and technicians on staff at our full-service, in-house repair center

**WE BUY USED EQUIPMENT**
Sell your excess, underutilized, and idle used equipment
We also offer credit for buy-backs and trade-ins
***www.artisantg.com/WeBuyEquipment*** ↗

***InstraView***™ **REMOTE INSPECTION**
Remotely inspect equipment before purchasing with our interactive website at ***www.instraview.com*** ↗

**LOOKING FOR MORE INFORMATION?**
Visit us on the web at ***www.artisantg.com*** ↗ for more information on price quotations, drivers, technical specifications, manuals, and documentation

***Contact us:*** (888) 88-SOURCE | sales@artisantg.com | www.artisantg.com