

ARTISAN®
TECHNOLOGY GROUP

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

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

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

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

**Custom engineering**
so your equipment works exactly as you specify.

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

**Expert team | Trust guarantee | 100% satisfaction**

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

**Visit our website - *Click HERE***

# Technical Reference Manual for PP 33x/02x CompactPCI® Intel® Pentium® M Processor Intelligent Dual PMC Carrier

Manual Order Code 560 0018 Rev 07 June 2009

**Concurrent Technologies Inc**
6 Tower Office Park
Woburn
MA 01801
USA
Tel: (781) 933 5900
Fax: (781) 933 5911

**Concurrent Technologies Plc**
4 Gilberd Court
Newcomen Way
Colchester, Essex CO4 9WN
United Kingdom
Tel: (+44) 1206 752626
Fax: (+44) 1206 751116

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

## NOTES

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

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

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

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

## CONVENTIONS

Throughout this manual the following conventions will apply:

- # or * after a name represents an active low signal. e.g. INIT* or INIT#
- h denotes a hexadecimal number. e.g. FF45h
- byte represents 8-bits
- word represents 16-bits
- dword represents 32-bits

## NOTATIONAL CONVENTIONS

| | |
|---|---|
| **NOTE:** | Notes provide general additional information. |

| | |
|---|---|
| **WARNING:** | Warnings provide indication of board malfunction if they are not observed. |

| | |
|---|---|
| **CAUTION:** | Cautions provide indications of board or system damage if they are not observed |

| | |
|---|---|
| ATA | AT Attachment |
| BIOS | Basic Input Output System |
| BMC | Baseboard Management Controller |
| CMOS | Complementary Metal Oxide Semiconductor |
| CPCI | CompactPCI |
| CPU | Central Processing Unit |
| CRT | Cathode Ray Tube |
| DDR | Double Data Rate |
| DIB | Dual Independent Bus |
| DIL | Dual In Line |
| ECC | Error Checking and Correcting |
| EIDE | Enhanced Integrated Drive Electronics |
| EMC | Electromagnetic Compatibility |
| EPROM | Electrically Programmable Read Only Memory |
| FRU | Field Replacement Unit |
| FWH | Firmware Hub |
| GMCH | Graphics Memory Controller Hub |
| ICH | I/O Controller Hub |
| IEEE | Institute of Electrical and Electronics Engineers |
| I/O | Input/Output |
| IPMB | Intelligent Platform Management Bus |
| IPMI | Intelligent Platform Management Interface |
| LED | Light Emitting Diode |
| LFM | Linear Feet per Minute |
| LPC | Low Pin Count |
| NMI | Non Maskable Interrupt |
| PC-AT | Personal Computer-Advanced Technology |
| PCI | Peripheral Component Interconnect |
| PICMG | PCI Industrial Computer Manufacturers Group |
| PMC | PCI Mezzanine Card |
| POST | Power-on Self Test |
| PRST | Push Button Reset |
| RST | Reset |
| RTC | Real Time Clock |
| SDR | Sensor Data Record |
| SDRAM | Synchronous Dynamic Random Access Memory |
| SMC | Satellite Management Controller |
| USB | Universal Serial Bus |

| Revision | Summary of Changes | Date |
|---|---|---|
| 01 | First Release | September 2004 |
| 02 | Added Sections 10.7 and 11.3 describing the Recovery BIOS and ECC Error logging; updated power supply current figures for all processor speeds; replaced 1.1 GHz CPU by 1.4 GHz | October 2004 |
| 03 | Minor corrections made throughout | November 2004 |
| 04 | Notes and conventions updated on page ii | December 2004 |
| 05 | Main Ethernet Controller type corrected from 82546EB to 82546GB; identified restricted baud rate on COM1 port | November 2005 |
| 06 | Added content for PCI 6254 CompactPCI bridge, change of Ethernet controllers and Rev D board | March 2006 |
| 07 | Minor corrections and added battery disposal warning | June 2009 |

1 INTRODUCTION ..... 1-1

1.1 General ..... 1-1

1.2 The PP 33x/02x - Main Features ..... 1-2
1.2.1 Central Processor ..... 1-2
1.2.2 Cache Memories ..... 1-2
1.2.3 Chipset ..... 1-2
1.2.4 SDRAM ..... 1-3
1.2.5 PCI Busses ..... 1-3
1.2.6 EPROM ..... 1-3
1.2.7 CompactFlash™ ..... 1-3
1.2.8 EIDE Controllers ..... 1-3
1.2.9 USB ..... 1-3
1.2.10 PMC Interfaces ..... 1-3
1.2.11 Ethernet Controllers ..... 1-3
1.2.12 CompactPCI Interface ..... 1-4
1.2.13 System Management ..... 1-4
1.2.14 Serial Communications ..... 1-4
1.2.15 Keyboard and Mouse ..... 1-4
1.2.16 Real Time Clock (RTC) ..... 1-4

1.3 AD PP5/001-3x Peripheral Functions ..... 1-5

1.4 Additional Board Options ..... 1-6

1.5 Extended Temperature Options ..... 1-7

1.6 Compliance to RoHS 2002/95/EC ..... 1-8

2 HARDWARE INSTALLATION ..... 2-1

2.1 General ..... 2-1

2.2 Unpacking and Inspection ..... 2-2

2.3 Default Jumper and Switch Settings ..... 2-3

2.4 Front Panel Indicators and Controls ..... 2-4
2.4.1 Run LED (R) Green ..... 2-4
2.4.2 POST LED (P) Yellow ..... 2-4
2.4.3 Ethernet Speed LEDs (SPD) Yellow ..... 2-4
2.4.4 Ethernet Link/Activity LEDs (LK/ACT) Green ..... 2-4
2.4.5 User LED (U) Red ..... 2-4
2.4.6 Hot-Swap LED (HS) Blue ..... 2-4
2.4.7 EIDE Activity LED (I) Yellow ..... 2-4
2.4.8 Switch (SW) ..... 2-5

2.5 On-board CompactFlash Site ..... 2-6

2.6 Installation of On-Board Mass Storage ..... 2-8
2.6.1 Hard Disk Storage Kit (AD CP1/DR1) ..... 2-9
2.6.2 CompactFlash Storage Kit (AD 200/001) ..... 2-10

2.7 Adding or Replacing DRAM Modules ..... 2-11

2.8 Battery Installation/Replacement ..... 2-12

2.9 Installing or Removing a PMC Module ..... 2-13
2.9.1 PMC Bus Speed Selection ..... 2-14

2.10 CompactPCI Operating Mode Selection ..... 2-15

2.11 Reset Sources ..... 2-16
2.11.1 CompactPCI Reset ..... 2-17

2.11.2 External Reset ..... 2-18
2.11.3 CompactPCI Push Button Reset ..... 2-19

2.12 Installation and Power-up ..... 2-20
2.12.1 Non Hot Swap Procedure - Installing ..... 2-21
2.12.2 Non Hot Swap Procedure - Removing ..... 2-21
2.12.3 Hot Swap Procedure - Installing ..... 2-21
2.12.4 Hot Swap Procedure - Removing ..... 2-22

**3 SOFTWARE INSTALLATION ..... 3-1**

3.1 Starting up for the first time ..... 3-1

3.2 Bootloading from CD-ROM ..... 3-2

3.3 Installing Microsoft® Windows® Operating Systems ..... 3-3

3.4 Installing RedHat® Linux® 8.0 and 9.0 ..... 3-4

**4 MASS STORAGE INTERFACES ..... 4-1**

4.1 EIDE Interfaces ..... 4-1

**5 ETHERNET INTERFACES ..... 5-1**

5.1 Rear Ethernet Configuration ..... 5-1

5.2 PICMG 2.16 Configuration ..... 5-1

**6 OTHER INTERFACES ..... 6-1**

6.1 Serial Ports ..... 6-1
6.1.1 PC BIOS Serial Console ..... 6-2
6.1.2 PC BIOS Serial Console Port ..... 6-3

6.2 Keyboard and Mouse Ports ..... 6-4

6.3 Graphics (VGA) Controller ..... 6-4

6.4 Real-Time Clock ..... 6-4

6.5 Universal Serial Bus (USB) ..... 6-4

6.6 Power On Self Test LED/Speaker ..... 6-4

**7 INTELLIGENT PLATFORM MANAGEMENT INTERFACE ..... 7-1**

7.1 Introduction ..... 7-1

7.2 IPMI Compatibility ..... 7-2

7.3 IPMI Overview ..... 7-3
7.3.1 Message Passing ..... 7-3
7.3.2 Events ..... 7-3
7.3.3 System Event Log ..... 7-3
7.3.4 Sensors ..... 7-4
7.3.5 Sensor Data Records ..... 7-4
7.3.6 Field Replaceable Unit Inventory Data ..... 7-4
7.3.7 Watchdog ..... 7-4

7.4 Supported Commands ..... 7-5
7.4.1 Get Self Test Results Command ..... 7-7
7.4.2 Watchdog Commands ..... 7-7
7.4.3 SEL and SDR Commands ..... 7-7
7.4.4 Sensor Commands ..... 7-7

7.4.4.1 Board Temperature Sensors ..... 7-8
7.4.4.2 Fan Monitor ..... 7-8
7.4.4.3 Board Voltage Sensors ..... 7-8
7.4.4.4 CompactPCI Slot Number ..... 7-8
7.4.4.5 CompactPCI BDSEL Signal ..... 7-8
7.4.4.6 CompactPCI SYSEN Signal ..... 7-8
7.4.4.7 CompactPCI PRESENT Signal ..... 7-8
7.4.4.8 Hot Swap Control Power Good ..... 7-8
7.4.4.9 Hot Swap Control Power Fail ..... 7-8
7.4.4.10 Ejector Handle ..... 7-9
7.4.4.11 Power Regulators Status ..... 7-9

7.5 FRU Inventory Data ..... 7-10
7.5.1 Common Header Area ..... 7-10
7.5.2 Board Area ..... 7-11
7.5.3 Board Area Format Version ..... 7-11
7.5.3.1 Board Area Length ..... 7-11
7.5.3.2 Language Code ..... 7-11
7.5.3.3 Manufacturing Date and Time ..... 7-11
7.5.3.4 Manufacturer's Name ..... 7-11
7.5.3.5 Board's Name ..... 7-11
7.5.3.6 Board's Port Number ..... 7-11
7.5.3.7 Serial Number ..... 7-11
7.5.3.8 FRU File ID ..... 7-11

7.6 Programming Examples ..... 7-12
7.6.1 Using the SMIC Interface ..... 7-12
7.6.2 Using the Watchdog Timer ..... 7-15
7.6.3 Reading Sensors ..... 7-18

**8 FLASH EPROM AND DRAM ..... 8-1**

8.1 Flash EPROM ..... 8-1

8.2 DRAM ..... 8-1

**9 ADDITIONAL LOCAL I/O FUNCTIONS ..... 9-1**

9.1 Status & Control Register 0 ..... 9-3

9.2 Status & Control Register 1 ..... 9-4

9.3 General Purpose I/O Register ..... 9-5

9.4 Status & Control Register 2 ..... 9-6

9.5 Interrupt Control Register ..... 9-7

9.6 Interrupt Configuration Register ..... 9-8

9.7 IPMI SMIC Interface ..... 9-9
9.7.1 SMIC Data Register ..... 9-9
9.7.2 SMIC Control/Status Register ..... 9-9
9.7.3 SMIC Flags Register ..... 9-9

9.8 P.O.S.T. LED / Speaker ..... 9-10

9.9 PORT 80 ..... 9-11

**10 PC BIOS ..... 10-1**

10.1 Entering the PC BIOS ..... 10-1

10.2 The PC BIOS Startup Sequence ..... 10-3

10.3 Boot Device Selection ..... 10-4

10.4 PCI Bus Resource Management ..... 10-5
10.4.1 PCI Resource Allocation ..... 10-5
10.4.2 PCI Device IDs ..... 10-7

10.5 CompactPCI Bridge Configuration ..... 10-8
10.5.1 System Controller Mode ..... 10-8
10.5.2 Satellite Mode ..... 10-8
10.5.3 Peripheral Mode ..... 10-8
10.5.3.1 Upstream Windows ..... 10-8
10.5.3.2 Downstream Windows ..... 10-8
10.5.4 Peripheral Mode Window-Size Limitations ..... 10-9
10.5.4.1 I/O Windows ..... 10-9
10.5.4.2 Memory Mapped Windows ..... 10-9

10.6 User Selectable NVRAM Defaults ..... 10-10

10.7 The Recovery BIOS ..... 10-12

**11 SYSTEM MANAGEMENT ..... 11-1**

11.1 Power Management ..... 11-1

11.2 Thermal Management ..... 11-2
11.2.1 Thermal Monitor 1 (TM1) ..... 11-2
11.2.2 Thermal Monitor 2 (TM2) ..... 11-2
11.2.3 CPU Thermal Trip ..... 11-2
11.2.4 Ethernet Controller Thermal Considerations ..... 11-2
11.2.5 PC BIOS Setup Options ..... 11-3
11.2.6 Processor Thermal Status Indication ..... 11-3

11.3 ECC Error Logging ..... 11-4
11.3.1 Single-bit ECC Errors ..... 11-4
11.3.2 Multi-bit ECC Errors ..... 11-4
11.3.3 Background ECC Event Logging ..... 11-4

**A SPECIFICATIONS ..... A-1**

A.1 Functional Specification ..... A-1

A.2 Environmental Specification (N Series) ..... A-2
A.2.1 Temperature Range ..... A-2
A.2.2 Humidity ..... A-2

A.3 Dimensions ..... A-3

A.4 Electrical Specification ..... A-4
A.4.1 Power Supply Requirements ..... A-4

A.5 Connectors ..... A-5
A.5.1 CompactPCI Interface (J1) Pin-outs ..... A-6
A.5.2 CompactPCI Interface (J2) Pin-outs ..... A-7
A.5.3 CompactPCI Interface (J3) Pin-outs ..... A-8
A.5.4 CompactPCI Interface (J5) Pin-outs ..... A-9
A.5.5 On-Board Mass Storage Option Connector (P6) Pin-outs ..... A-10
A.5.6 PMC Site Connectors (J11 - J14 and J21 - J24) Pin-outs ..... A-11
A.5.7 Ethernet Connectors (P4 and P5) Pin-out ..... A-15
A.5.8 Port 80 (S2) Pin-outs ..... A-16
A.5.9 USB Connector (P2 and P3) Pin-outs ..... A-17
A.5.10 Serial Port 10-way (LK4) IDC Connector ..... A-18

# TABLE OF FIGURES


Figure 1-1 Overview ..... 1-1
Figure 2-1 Default Jumper and Switch Settings ..... 2-3
Figure 2-2 Front Panel Indicators and Controls ..... 2-4
Figure 2-3 Front Panel Switch Function Switch ..... 2-5
Figure 2-4 CompactFlash Site ..... 2-6
Figure 2-5 CompactFlash Master/Slave Switch ..... 2-7
Figure 2-6 Mass Storage Connector and Fixing Holes ..... 2-8
Figure 2-7 Disk Drive Cable Installation ..... 2-9
Figure 2-8 CompactFlash Carrier Module Installation ..... 2-10
Figure 2-9 AD 200/001 DIL Switch Settings ..... 2-10
Figure 2-10 DRAM Module Replacement ..... 2-11
Figure 2-11 Battery Fitting and CMOS Clear Jumper ..... 2-12
Figure 2-12 PMC Installation Diagram ..... 2-13
Figure 2-13 PMC Bus Speed Select Switch ..... 2-14
Figure 2-14 Satellite Mode Switch ..... 2-15
Figure 2-15 CompactPCI Reset Input Switch ..... 2-17
Figure 2-16 External Reset Switch ..... 2-18
Figure 2-17 CompactPCI Push Button Reset Switch ..... 2-19
Figure 6-1 Console Switch ..... 6-2
Figure 6-2 Console Port Switch ..... 6-3
Figure 10-1 Boot Mode Switch ..... 10-1
Figure 10-2 BIOS Defaults Switch ..... 10-10
Figure 10-3 Boot Type Switch ..... 10-12

Figure A-1 Connector Layout ..... A-5
Figure A-2 Front Panel Connectors ..... A-5
Figure A-3 USB Connectors ..... A-17

# TABLE OF TABLES


Table 2-1 Reset Configuration Options....2-16
Table 6-1 Serial Port Numbering....6-1
Table 7-1 Supported IPMI Commands....7-6
Table 7-2 Sensor ID Codes....7-7
Table 7-3 FRU Inventory Area : Common Header Area Data....7-10
Table 7-4 FRU Inventory Area : Board Area Data....7-11
Table 9-1 I/O Address Map....9-1
Table 10-1 Configurable PCI Bus Interrupts....10-6
Table 10-2 PCI Device Numbers....10-7
Table 11-1 CPU Operating Speed BIOS Options....11-1

Table A-1 Voltage and Current Requirements....A-4
Table A-2 CompactPCI J1 Interface Pin-outs....A-6
Table A-3 CompactPCI J2 Interface Pin-outs....A-7
Table A-4 CompactPCI J3 Interface Pin-outs....A-8
Table A-5 CompactPCI J5 Interface Pin-outs....A-9
Table A-6 On-Board Mass Storage Option Interface Pin-outs....A-10
Table A-7 PMC J11 and J21 Connector Pin-outs....A-11
Table A-8 PMC J12 and J22 Connector Pin-outs....A-12
Table A-9 PMC J13 and J23 Connector Pin-outs....A-13
Table A-10 PMC J14 and J24 Connector Pin-outs....A-14
Table A-11 Ethernet RJ-45 Connector Pin-outs....A-15
Table A-12 Port 80 Connector Pin-outs....A-16
Table A-13 USB Connector Pin-outs....A-17
Table A-14 COM1 LK4 Serial Port 10-way IDC Connector....A-18

# 1 INTRODUCTION

## 1.1 General

This manual is a guide and reference handbook for engineers and system integrators who wish to use the Concurrent Technologies' PP 33x/02x ultra high-performance Intel® Pentium® M processor single board computer. The board has been designed for high-speed multiprocessing applications using a PC-AT™ architecture operating in a CompactPCI Bus environment.

The PP 33x/02x board is available in several different variants which differ by the processor speed, the amount of fitted SDRAM and the rear Ethernet connector configuration. Currently the board is available with a 1.6 GHz Intel® Pentium® M processor, a 1.4 GHz Intel® Pentium® M processor Low Voltage 738 or a 1.8 GHz Intel® Pentium® M processor 745. These models are designated PP 330/022, PP 332/020 and PP 332/021 respectively. The other configuration options are specified by a two-digit suffix to the board name; refer to the product data sheet for further details. Further details of other board options are given in Section 1.3. References to the board in this document will use the name PP 33x/02x unless they apply only to a specific variant, in which case the full name will be used.

The information contained in this manual has been written to provide users with all the information necessary to configure, install and use the PP 33x/02x as part of a system. It assumes that the user is familiar with the CompactPCI bus and PC-AT bus architectures and features.

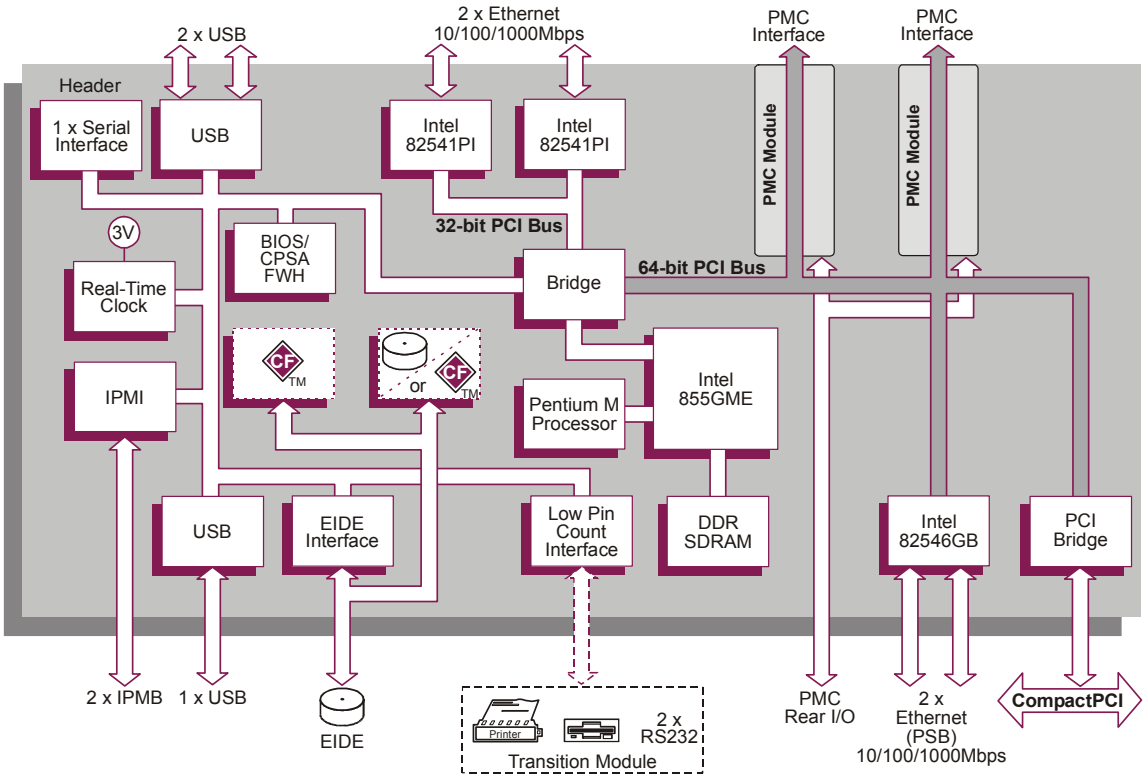



**Figure 1-1 Overview**

## 1.2 The PP 33x/02x - Main Features

The PP 33x/02x is a member of the Concurrent Technologies range of single-board computers for the CompactPCI bus architecture. It has been designed as a powerful single board computer based upon the Intel Pentium M Processor, the Intel® 855GME and 6300ESB chipset, an 82546GB dual channel Ethernet controller and two 82541PI single channel Ethernet controllers. It also provides two IEEE 1386.1 PMC interfaces, optional on-board mass storage, and interfaces for standard PC-AT based peripherals. Elsewhere in this manual the processor is referenced as the Intel Pentium M processor regardless of the exact CPU chip being used.

**NOTE:** Earlier revisions of the board were fitted with Intel 82540EM Gigabit Ethernet controllers instead of the 82541PI controllers. References to the Intel 82541PI in this manual also apply to the 82540EM unless otherwise stated.

### 1.2.1 Central Processor

The central processor used on this board is an ultra high performance Intel Pentium M processor, operating internally at 1.4 GHz, 1.6 GHz or 1.8 GHz. The processor supports the Dual Independent Bus (DIB) architecture with the backside bus connected to the on die Level 2 cache and the 64-bit frontside bus connected to the memory controller at 400 MHz to provide a maximum transfer bandwidth of 3.2 Gbytes/s. The processor is capable of addressing 64 Gbytes of physical memory all of which is cacheable, and 64 Terabytes of virtual memory. The Intel Pentium M processor is upwardly code-compatible with the other members of the x86 family of microprocessors.

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

### 1.2.2 Cache Memories

The Level 1 and Level 2 caches are both implemented on the processor die for maximum performance. The Level 1 cache stores 32 Kbytes of instructions and 32 Kbytes of data. The Level 2 cache on the 1.6 GHz processor stores 1 Mbyte of instructions and data. The Level 2 cache on the 1.4 GHz and 1.8 GHz processors stores 2 Mbytes of instructions and data.

The Level 2 cache operates at the core frequency and is based on Intel's Advanced Transfer Cache architecture.

### 1.2.3 Chipset

The PP 33x/02x uses the Intel 855GME and 6300ESB chipset. This is comprised of the 855GME Graphics Memory Controller Hub (GMCH) and the 6300ESB I/O Controller Hub (ICH). They respectively provide North Bridge + Graphics and South Bridge + PCI bus controller functionality.

The 855GME GMCH interfaces to the CPU's host bus. It provides a DDR SDRAM memory controller and a high speed Hub Interface 1.5 which connects to the 6300ESB ICH. It supports concurrent CPU, memory and Hub Interface bus operations.

The 6300ESB ICH provides a 66 MHz 64-bit PCI or PCI-X bus for supporting high performance PCI devices. It also provides a 33 MHz 32-bit PCI bus. The Hub Interface supports a maximum transfer bandwidth of 266 Mbytes/s.

The 6300ESB ICH also provides a variety of peripheral functions including Serial ATA (SATA) controllers, EIDE controllers, USB 2.0 and 1.1 controllers, LPC (Low Pin Count) Bus bridge, IOAPIC interrupt controller, Real Time clock (RTC), two serial ports and other legacy PC-AT architectural functions.

The LPC Bus is used to connect to the Firmware Hub (FWH), the control and status registers and to the PC87391 Super I/O Controller on the companion AD PP5/001-3x Transition Module. This device implements the floppy disk controller, the parallel port and two additional serial ports.

### 1.2.4 SDRAM

The on-board SDRAM controller provides a DDR266 or DDR333 channel to provide a maximum transfer bandwidth of 2.7 Gbytes/s and features ECC data protection. Two 200-pin SODIMM sockets are provided. These accept standard unbuffered PC2100 or PC2700 DDR SDRAM modules having a capacity up to 1 Gbyte each. Hence a maximum of 2 Gbytes of SDRAM may be fitted to the board. Both ECC and non-ECC modules are supported.

### 1.2.5 PCI Busses

There are two on-board PCI busses supported by the 6300ESB ICH. The primary bus is 64-bits wide and provides a high performance, up to 528 Mbytes/s, connection between the 6300ESB ICH, 82546GB Ethernet controller, PMC sites and the CompactPCI bridge. The secondary PCI bus is 32-bits wide and provides a lower performance, up to 132 Mbytes/s, connection between the 6300ESB ICH and the two 82541PI Ethernet controllers.

### 1.2.6 EPROM

The board contains a 1 Mbyte Firmware Hub (FWH) for the BIOS code and fixed data and factory test software.

### 1.2.7 CompactFlash™

The PP 33x/02x contains an on-board Compact Flash site, which is located below PMC site 1. The Compact Flash site primarily supports Type 1 modules. Type 2 modules may be fitted subject to them not interfering mechanically with the PMC module. Compact Flash provides a flexible and cost-effective alternative to on-board Flash memory for application program and data storage.

### 1.2.8 EIDE Controllers

The PP 33x/02x has two EIDE/Ultra ATA100 interfaces. One EIDE interface is available via the J5 connector, the other via an on-board connector for use by the Compact Flash site and the optional on-board disk drive or dual CompactFlash carrier.

### 1.2.9 USB

The PP 33x/02x provides three USB channels, two via the front panel and a third via the CompactPCI J5 connector. All three USB channels support both USB 1.1 (1.5 Mbit/s and 12 Mbits/s) and USB 2.0 (480 Mbits/s) operation.

### 1.2.10 PMC Interfaces

Two PMC interfaces which support single width 64 or 32-bit PMC modules complying with the IEEE 1361.1 standard are provided. A 3.3V PCI signaling environment is supported.

The PMC interfaces will also accept Processor PMC modules. The latter will operate only in non-Monarch mode.

### 1.2.11 Ethernet Controllers

An Intel® 82546GB dual channel Gigabit Ethernet controller is used to provide high performance PCI to Ethernet interfaces. Both channels support 10, 100 and 1000Mbits/s operation and are routed to the CompactPCI J3 connector.

The board can either support PICMG® 2.16 backplane networking or rear panel Ethernet. This is specified by an ordering option. A suitable Transition Module, such as the AD PP5/001-3X, is required for rear panel Ethernet.

The PP 33x/02x also provides two front panel Ethernet interfaces. These are implemented with two Intel® 82541PI Gigabit Ethernet controllers and both support 10, 100 and 1000 Mbits/s operation.

### 1.2.12 CompactPCI Interface

The PP 33x/02x is a CompactPCI compatible System Controller board. It may also act as a Peripheral board or as a Satellite board in any backplane slot. The board uses a 64-bit interface implemented with a PLX™ Technology PCI 6254 PCI to PCI bridge. As a System Controller it supports PICMG 2.1 compliant Hot Swap Peripheral Boards. As a Peripheral board it may be Hot Swapped under the control of the System Controller. As a Satellite board it may be Hot Swapped under the control of an on-board microcontroller.

**NOTE:** Some variants of the Rev D board may be fitted with a PLX Technology PCI 6154 PCI to PCI bridge. This bridge does not support Peripheral board operation. Earlier revisions of the board were usually fitted with this bridge.

### 1.2.13 System Management

The PP 33x/02x supports PICMG 2.9 system management. The IPMI SMIC interface is implemented and the CompactPCI IPMB 0 and IPMB 1 communication channels are provided. An on-board microcontroller provides the requisite IPMI functionality. The microcontroller can be configured to provide Baseboard Management Controller (BMC) functionality.

### 1.2.14 Serial Communications

The PP 33x/02x provides one fixed RS232 serial data communication port. This is implemented by the 6300ESB ICH and is available via an in-board 10-way header.

**NOTE:** Two more serial ports are provided on the AD PP5/001-3x Transition Module. These ports are not available if the second serial port option referred to above is fitted.

### 1.2.15 Keyboard and Mouse

The PP 33x/02x supports USB keyboard and mouse via the two front panel USB interfaces. PS/2™ type keyboard and mouse interfaces are not supported.

### 1.2.16 Real Time Clock (RTC)

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

## 1.3 AD PP5/001-3x Peripheral Functions

The AD PP5/001-3x Transition Module is the matching product for the PP 33x/02x. The AD PP5/001-3x contains a PC87391 Super I/O controller. This device provides the following interfaces:

- Floppy disk interface for up to two floppy disk drives
- Parallel port interface
- Two RS232 serial interfaces

**NOTE:** The Super I/O controller, and hence the above interfaces, are not available if the second serial port option via J5 is fitted.

The AD PP5/001-3x also provides:

- Two general purpose inputs
- Two general purpose outputs
- External Reset input
- Fan sensor input

The AD PP5/001-3x also contains either a CompactFlash site (AD PP5/001-30) or an on-board 1.8 inch hard disk drive (AD PP5/001-31). Only one of these options may be fitted at a time. Refer to the AD PP5/001 datasheet for ordering information.

## 1.4 Additional Board Options

The PP 33x/02x board may be ordered with one of a few factory fitted configuration options, in particular the Ethernet (on J3 connector):

- Configured for PICMG 2.16 backplane networking
- Configured for rear panel Ethernet via connectors (on Transition Module)

Two mezzanine mass storage options are available, namely:

- A 2.5 inch EIDE hard disk drive of at least 10 Gbyte capacity
- A dual CompactFlash carrier that also supports the Hitachi GST MicroDrive™

Refer to the PP 33x/02x datasheet for ordering information, which also details the SDRAM capacity options.

**NOTE:** The mass storage options use the PMC site 1 area. No PMC module may be fitted in this site when a mass storage option is fitted. PMC site 2 is not affected by fitting a mass storage option.

## 1.5 Extended Temperature Options

All variants of the board are qualified for the standard operating and storage temperature ranges indicated in Section A.2.1. Some variants of the board are available which offer a wider range of operating and storage temperatures, but certain board features are no longer available with these variants. In particular the option for on-board mass storage using a 2.5 inch hard disk drive is not supported for the extended temperature specifications.

Consult your distributor or Concurrent Technologies directly for details of the extended temperature options.

Several restrictions apply to Gigabit operation of the Ethernet interfaces on extended temperature boards. They are intended to prevent overheating of the Ethernet controllers.

- The rear Ethernet interfaces should not be operated at Gigabit speed if PMC modules are fitted and the ambient temperature exceeds 70°C.
- The front panel Ethernet interfaces should not be operated at Gigabit speed if the ambient temperature exceeds 70°C.
- The front panel Ethernet interfaces should not be operated at Gigabit speed if a bottom side cover is fitted and the ambient temperature exceeds 55°C.

## 1.6 Compliance to RoHS 2002/95/EC

This product is offered in a lead free (Pb) form which complies with the RoHS 2002/95/EC directive. The European Union RoHS 2002/95/EC directive restricts the use of six materials in electronic components and assemblies. Specifically, these materials are Lead (Pb), Mercury (Hg), Cadmium (Cd), Hexavalent Chromium (Cr VI), Polybrominated Biphenyls (PBB) and Polybrominated Diphenyl Ethers (PBDE). Concurrent Technologies is committed to compliance to the RoHS directive.




The product may also be available in an alternative form which includes components which are not lead-free or do not comply with the RoHS directive. Consult your supplier to determine order codes for RoHS compliant and non-compliant products.

# 2 HARDWARE INSTALLATION

## 2.1 General

This chapter contains general information on unpacking and inspecting the PP 33x/02x after shipment, and information on how to configure board options and install the board into a CompactPCI chassis.

**CAUTION:** It is strongly advised that, when handling the PP 33x/02x and its associated components, the user should at all times wear an earthing strap to prevent damage to the board as a result of electrostatic discharge.

The list below outlines the steps necessary to configure and install the board. Each entry in the list refers to a section in this chapter which will provide more details of that stage of the procedure. It is recommended that the board is configured in the sequence below.

1) Unpack the board - see Section 2.2.
2) Locate and familiarize yourself with the front panel indicators and controls - see Section 2.4.
3) Check that the board jumper settings match the required operating mode - see Section 2.3 for details of the default settings and where to find more information.
4) Fit the battery if necessary - see Section 2.8.
5) Fit a Compact Flash module if required - see Sections 2.5 and 2.6.
6) Fit the PMC modules(s) if required - see Section 2.9.
7) Configure the board for the correct CompactPCI operating mode and for any external sources for board or system resets - see Sections 2.10 and 2.11.
8) Install the board, using Hot Swap or non-Hot Swap procedures - see Section 2.12.

# HARDWARE INSTALLATION

## 2.2 Unpacking and Inspection

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

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

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

## 2.3 Default Jumper and Switch Settings

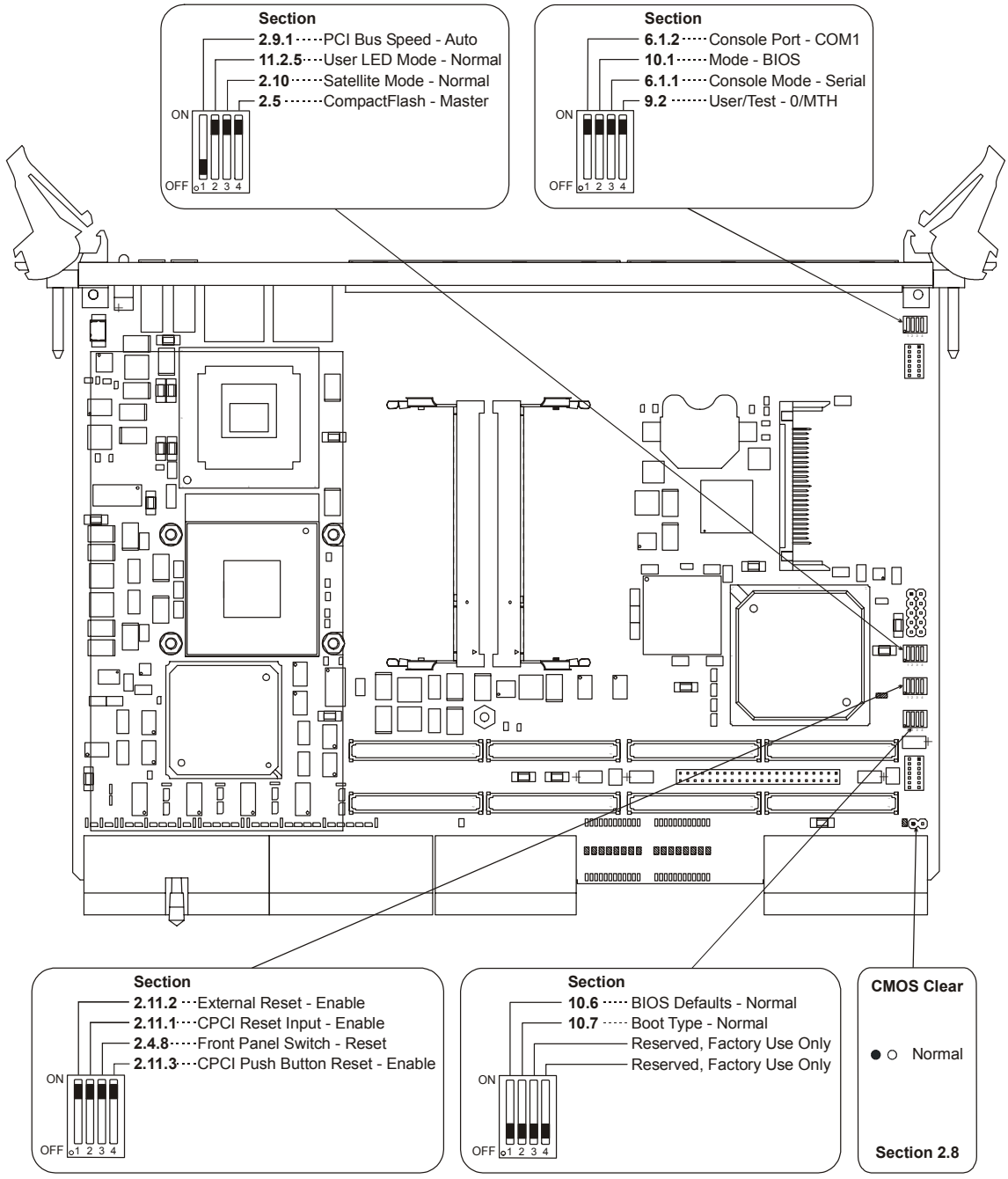



**Figure 2-1 Default Jumper and Switch Settings**

## 2.4 Front Panel Indicators and Controls

When installing or removing the board for the first time, or when checking its operation, it can be very useful to note the behavior of the LEDs on the front panel. Figure 2-2 shows the location of the LEDs, and their purpose is outlined below.

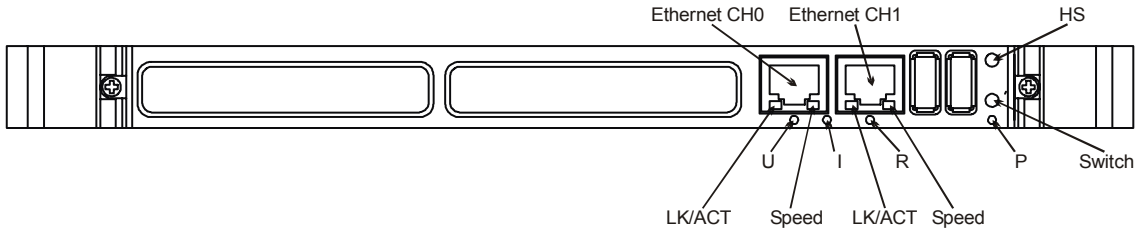



**Figure 2-2 Front Panel Indicators and Controls**

### 2.4.1 Run LED (R) Green

The run LED indicates that activity is occurring on the LPC bus or 32-bit PCI bus. This allows the user to quickly assess how active the busses are.

### 2.4.2 POST LED (P) Yellow

The POST LED is used to indicate that a power on self test has failed. This LED will also flash when outputting sound on the speaker.

### 2.4.3 Ethernet Speed LEDs (SPD) Yellow

These LEDs indicate the operating speed of the front panel Ethernet interfaces, as follows:

- Off = 10Mbits/s
- Steady On = 100Mbits/s
- Flashing = 1000Mbits/s

### 2.4.4 Ethernet Link/Activity LEDs (LK/ACT) Green

These LEDs light when connection has been made on the corresponding Ethernet interfaces. They will flash to indicate link activity, and during periods of high Ethernet activity the LED may switch off for several seconds.

### 2.4.5 User LED (U) Red

This LED is available for use by user software. It is manipulated via an I/O register (see Section 9.1 for details). The User LED may flash rapidly if the CPU Thermal Trip activates (see Section 11.2.5).

### 2.4.6 Hot-Swap LED (HS) Blue

This LED is used when the board is hot-swapped, and lights to indicate when the board can be safely removed from the chassis.

### 2.4.7 EIDE Activity LED (I) Yellow

This LED lights when there is activity on the on-board (secondary) EIDE interface.

**NOTE:** The AD PP5/001-3x Transition Module contains a LED to indicate activity on the off-board (primary) EIDE interface.

### 2.4.8 Switch (SW)

A pushbutton switch is recessed behind the front panel, and provides a means of generating a reset or NMI to the board.

The reset or NMI function is selected by the setting of the Front Panel Switch Function switch shown in Figure 2-3.

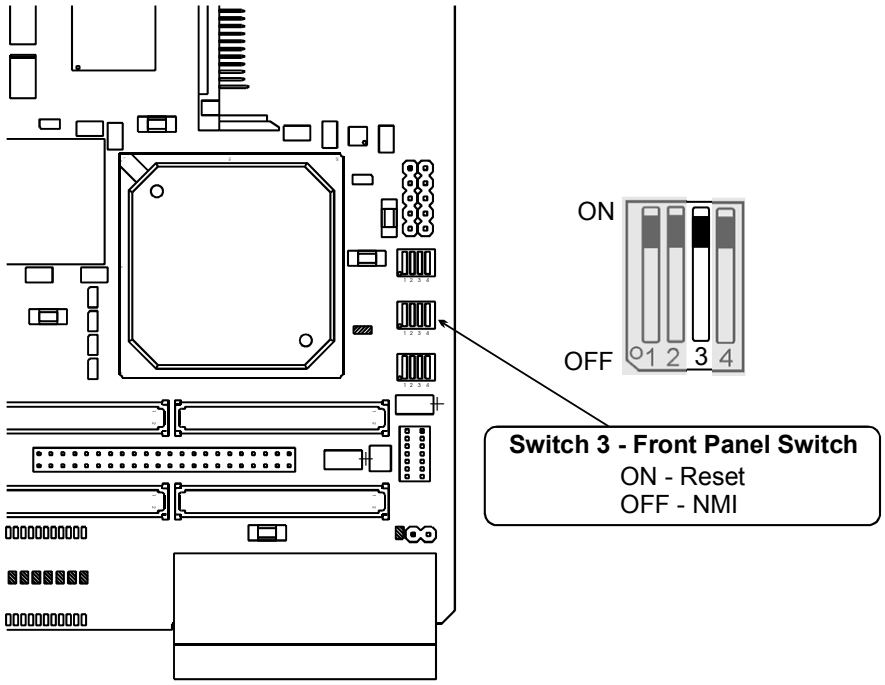



**Figure 2-3 Front Panel Switch Function Switch**

Selecting the Reset position setting will cause the board to be reset when the front panel switch is operated. If the board is in the System Controller Slot, it will also assert RST# on the CompactPCI backplane and hence reset the other boards in the chassis. If the board is operating as a Peripheral or Satellite, it will respond to front panel resets but will not reset the other boards in the chassis.

Selecting the NMI position setting configures the switch to generate NMI when operated. No reset is generated in this case.

## 2.5 On-board CompactFlash Site

A CompactFlash site is provided on the PP 33x/02x board. The site fully supports Type I modules. A Type II module or a MicroDrive may be fitted, but the user should be aware that these are 5.0mm tall and therefore encroach into the clearance zone between the PP 33x/02x board and PMC module. This may cause an interference problem if the PMC module also encroaches into this clearance zone.

The CompactFlash site is located under PMC site 1 as shown in Figure 2-4 below. The site faces the top edge of the board, hence the board must be removed from the chassis in order to insert or remove the module. To fit a module, orient it such that the connector end is facing the socket and the label side is away from the board, and gently slide it into the socket.

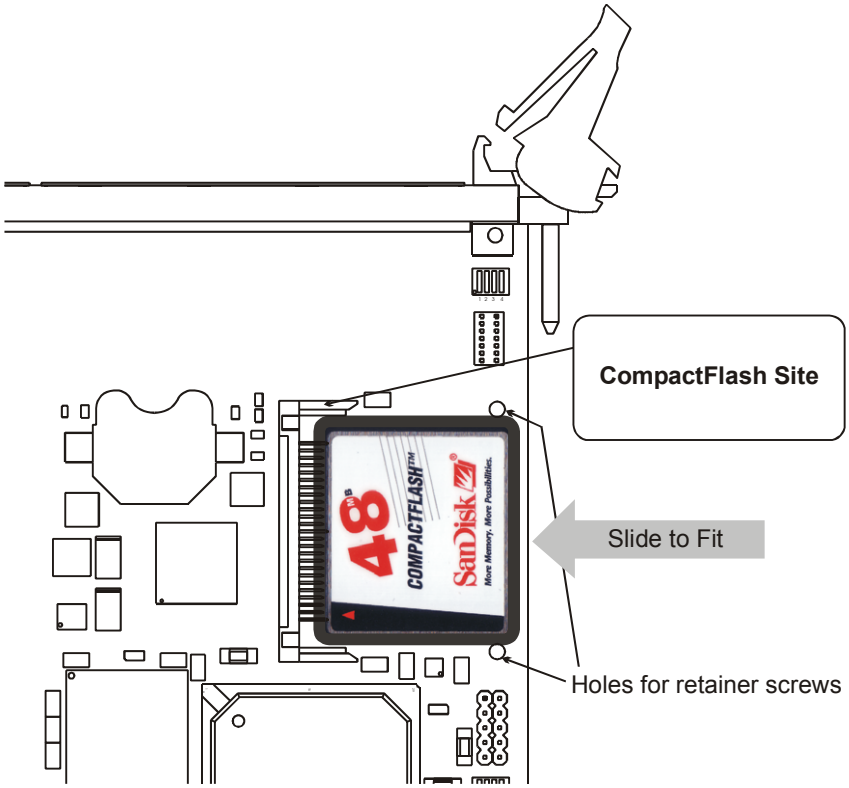



**Figure 2-4 CompactFlash Site**

The module will normally be adequately retained by the card edge guide of the chassis. However, holes are provided for a retaining strap, part no KT BKT/001-00.

To fit the strap, place it over the module, insert the two screws supplied with the strap from the bottom side of the board and tighten the screws. Do not over tighten the screws.

**NOTE:** If the board is likely to be subject to mechanical vibration a suitable thread lock compound applied to the screws should be considered.

The CompactFlash socket is connected to the secondary EIDE interface. A switch is provided to select whether the CompactFlash module is the IDE Master or Slave device. The switch settings are shown in Figure 2-5 below.

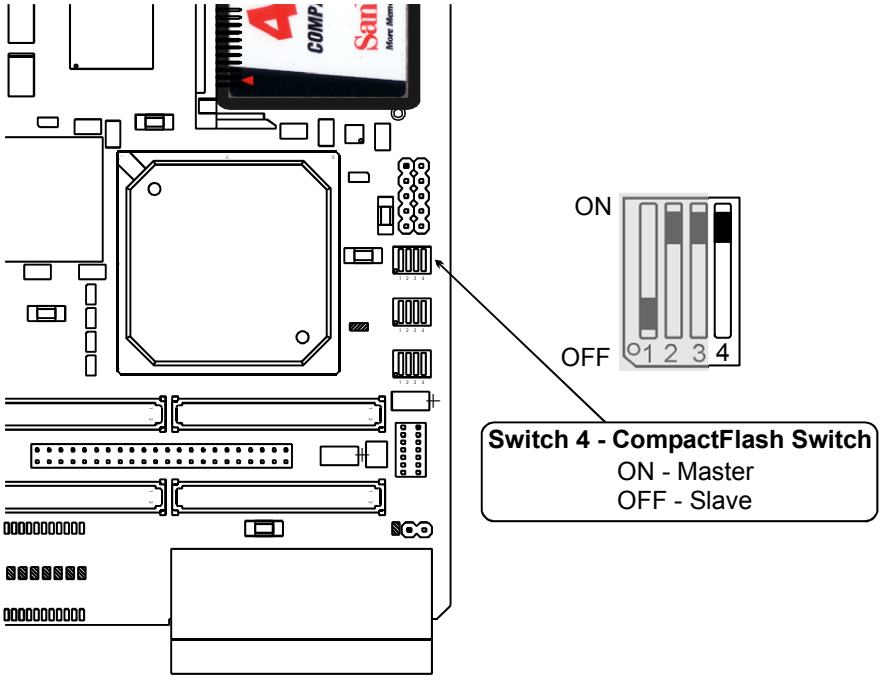



**Figure 2-5 CompactFlash Master/Slave Switch**

If the CompactFlash module is to be the only mass storage device fitted to the board, the switch should be set to the Master position.

If the CompactFlash site is to be used in conjunction with the AD CP1/DR1 Hard Disk Storage Kit, the switch is best set to the Slave position. The reason for this is that the disk drive is normally shipped configured as a Master device. However, some CompactFlash modules will not operate as IDE Slave devices. If you have such a module, set the switch to the Master position and configure the disk drive to be the Slave device. (The label on the drive will have instructions on how to do this).

The On-board CompactFlash site can also be used in conjunction with the AD 200/001 CompactFlash Mass Storage Kit (see Section 2.6.2), but only one module can then be fitted to the AD 200/001. The Master/Slave switch can be set to either position. If the switch is set to the Master position, then only Site 2 on the AD 200/001 can be used. Conversely, if the switch is set to the Slave position, then only Site 1 on the AD 200/001 can be used.

## 2.6 Installation of On-Board Mass Storage

If an on-board mass storage option has been ordered, it will be necessary to install the option at this time.

The mass storage option plugs into connector P6 and is secured via screws and spacers using the four mounting holes as shown in Figure 2-6 below.

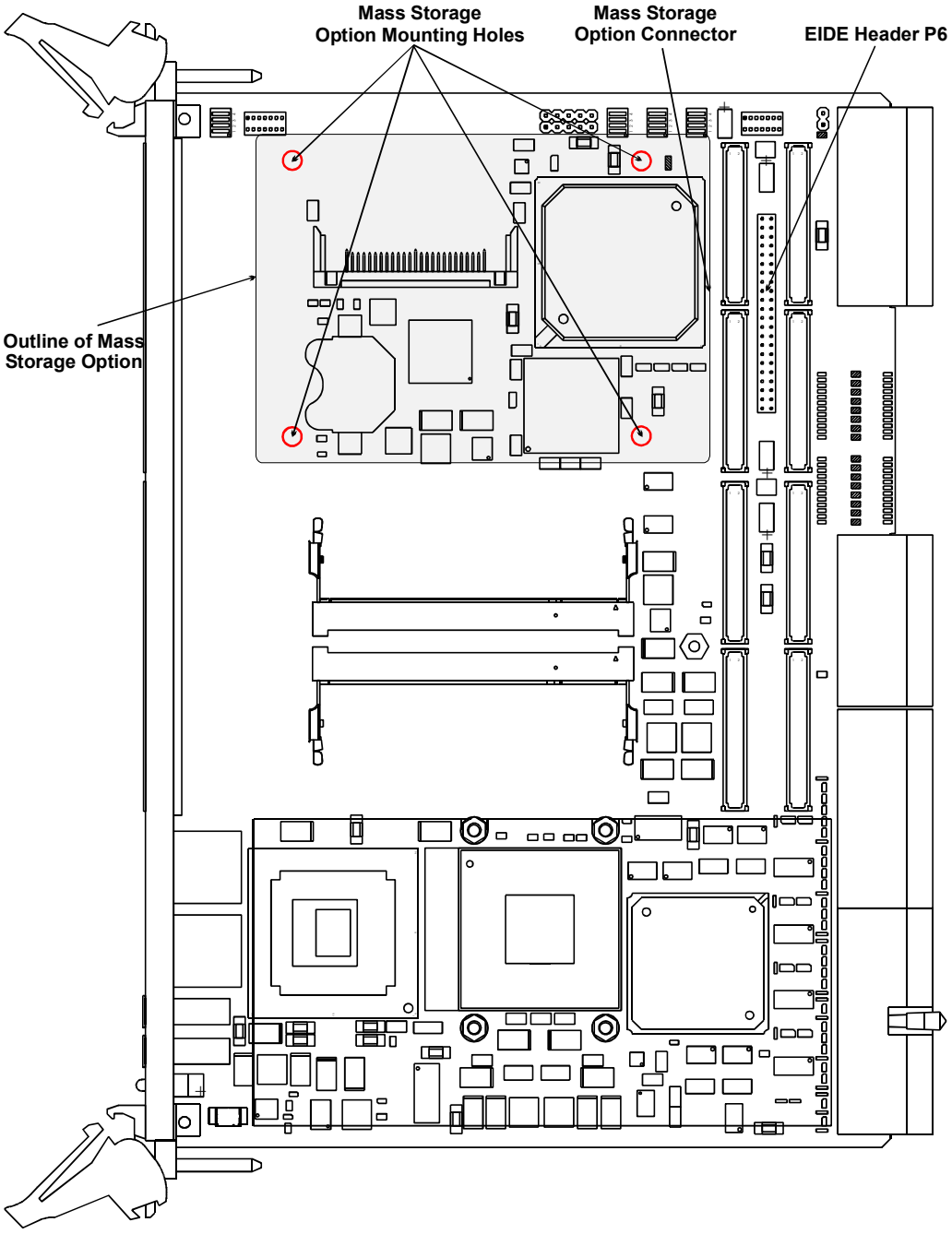



**Figure 2-6 Mass Storage Connector and Fixing Holes**

### 2.6.1 Hard Disk Storage Kit (AD CP1/DR1)

The option kit comprises:

- A 2.5 inch EIDE disk drive
- A ribbon cable assembly
- Four M3 x 10mm screws
- Four M3 x 5mm spacers

The ribbon cable assembly has a 50-way connector at one end and a 44-way connector at the other end. The 50-way connector plugs into the disk drive and the 44-way plugs into P6 on the PP 33x/02x.

9) Plug the 50-way connector into the disk drive as shown in Figure 2-7 below, note the orientation.

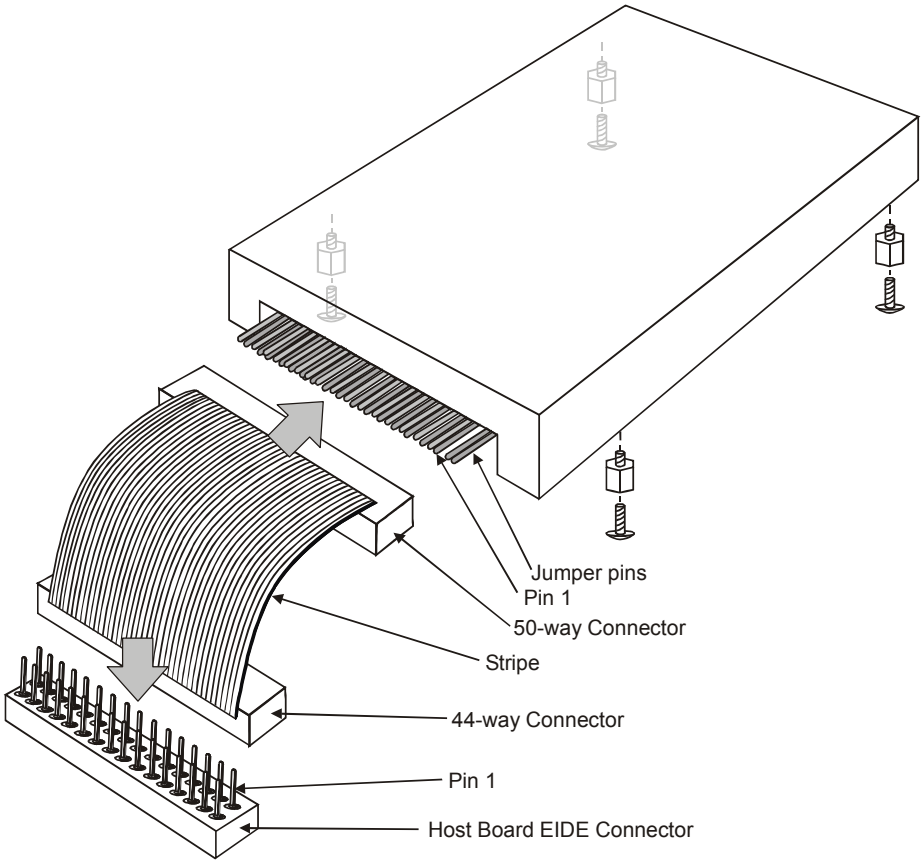



**Figure 2-7 Disk Drive Cable Installation**

10) Plug the 44-way connector into P6, note the orientation.
11) Fix the disk drive into position using the four screws and spacers provided. Do not over tighten the screws.

**NOTE:** If the board is likely to be subjected to mechanical vibration a suitable thread lock compound applied to the screws should be considered.

### 2.6.2 CompactFlash Storage Kit (AD 200/001)

The option kit comprises:

- A CompactFlash carrier module with attached ribbon cable
- Four M3 panhead screws

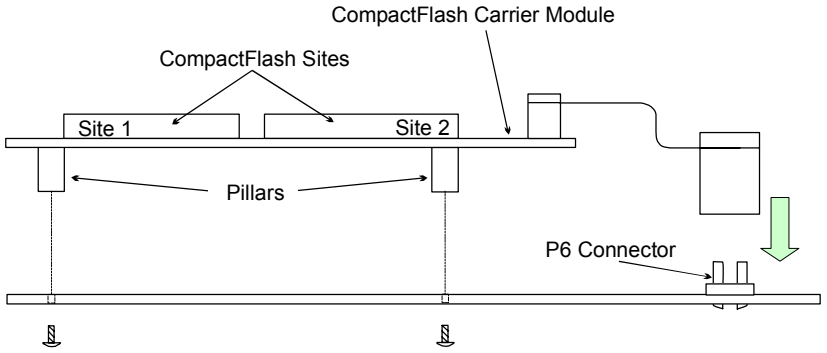



**Figure 2-8 CompactFlash Carrier Module Installation**

To install the Compact Flash carrier module, follow these instructions:

1) The M3 panhead screws may be loosely screwed into the end of the pillars, if so unscrew them.

**NOTE:** Do not unscrew the countersunk screws attaching the pillars to the circuit board.

2) Fix the module into position using the four panhead screws referred to earlier. Do not over tighten the screws.

**NOTE:** If the board is likely to be subjected to mechanical vibration a suitable thread lock compound applied to the screws should be considered.

3) Connect the captive ribbon cable connector to P6 on the PP 33x/02x board.

The CompactFlash sites are labeled CompactFlash 1 and CompactFlash 2.

If a single CompactFlash card is fitted, it should always be installed into site 1. Site 2 should be used only when two CompactFlash cards are fitted.

The CompactFlash card(s) may be retained in position by fitting short M3 screws and spacers into the holes near the long edge of the carrier. This will protect against accidental removal due to vibration or deliberate but unauthorized removal.

**NOTE:** If more than one CompactFlash module is fitted, the module in the CompactFlash 2 site must support operation as a Slave device.

The DIL switch on the AD 200/001 should be set as shown in Figure 2-9.

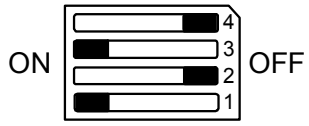



**Figure 2-9 AD 200/001 DIL Switch Settings**

## 2.7 Adding or Replacing DRAM Modules

The PP 33x/02x accepts standard 200-pin DDR SODIMM modules fitted with 2.5V PC2100 or PC2700 DDR SDRAM. Both 64-bit non-ECC SODIMM modules and 72-bit ECC SODIMM modules are supported. If ECC is required, then ECC modules **must** be fitted. Two sockets are provided and will accommodate modules up to 1 Gbyte capacity each.

**NOTE:** SODIMMs using DRAMs with 8K refresh are required.

Figure 2-10 shows the way in which SODIMMs are fitted or removed. When fitting only one SODIMM this must be in the SODIMM #1 position. No other changes are necessary when a SODIMM is added or removed.

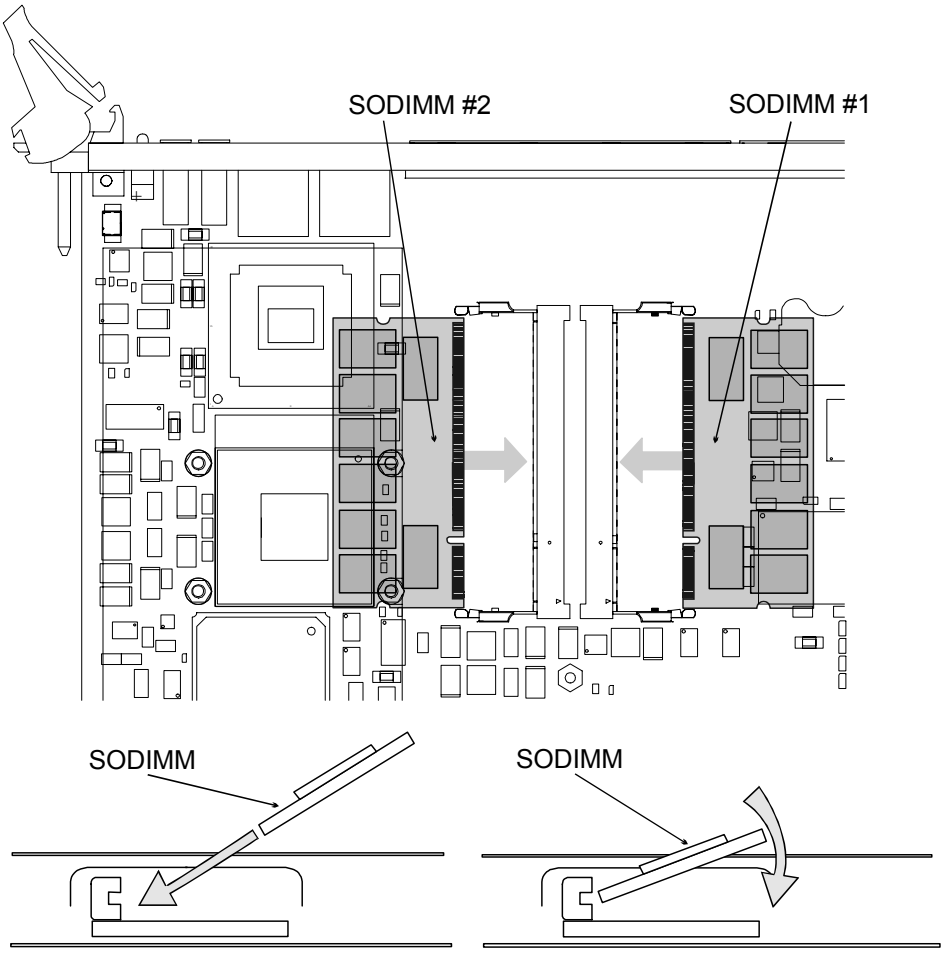



**Figure 2-10 DRAM Module Replacement**

## 2.8 Battery Installation/Replacement

The on-board Real-Time Clock and CMOS memory used by the PC BIOS firmware are powered by a 3.3V Lithium battery when the board is powered OFF. It is advisable, for the battery to be fitted prior to using the board. Figure 2-11 shows how to do this. One battery is supplied with the board, but it is not normally fitted.

If the board is operated without a battery, the User Selectable NVRAM Defaults feature can be used to override the factory default NVRAM settings. See Section 10.6 for further details.

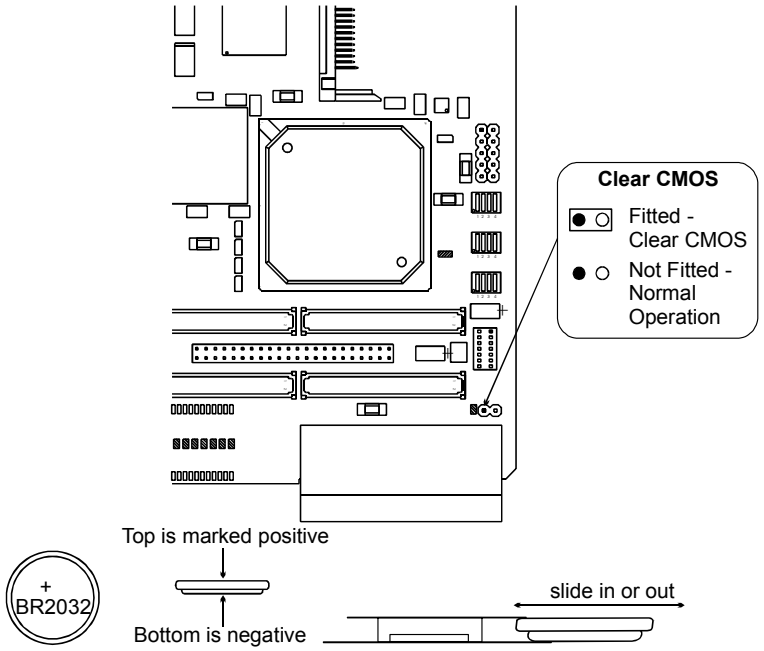



**Figure 2-11 Battery Fitting and CMOS Clear Jumper**

The battery should be replaced when the voltage falls below 2.8V. Depending on the way in which the board is operated and stored, battery life should be in excess of 5 years. The life expectancy will fall if the battery is subjected to long periods at temperatures of 45°C or above. It will also fall if the battery is fitted to a board that is stored in its conductive bag even at room temperature.

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

**WARNING:** The date and time settings will need to be initialized if the battery is disconnected.

**Battery**

Never dispose of batteries in a fire.
Always dispose of batteries according to local regulations.
Please recycle when possible.
Do not dispose of batteries as household waste.
Always replace battery with manufacturers recommended type.

If the BIOS setup screens have been used to set up the board for an invalid configuration, or in other fault conditions, it may be useful to be able to reset the contents of the CMOS RAM and Real-Time Clock. In this case, the CMOS Clear Jumper can be used.
To clear the CMOS RAM to a known state, fit the CMOS Clear jumper and apply power. When the board is next powered down remove the jumper, otherwise CMOS RAM will again be reset.

## 2.9 Installing or Removing a PMC Module

The PP 33x/02x boards provides 3.3V V(I/O) to the PMC sites. Before installing a PMC module check that it supports 3.3V V(I/O) If two PMC modules are fitted, their V(I/O) requirements must be the same.

**CAUTION:** Do not fit PMC modules designed for 5V V(I/O) only. Doing so may cause damage to the module or the PP 33x/02x.

**NOTE:** The PP 33x/02x board is supplied with PMC blanking plates which must be removed before a new module can be fitted.

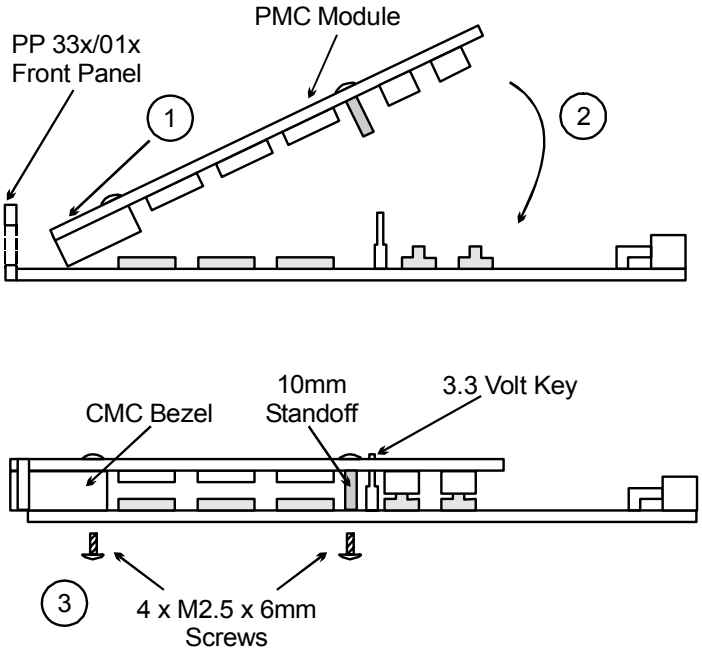



**Figure 2-12 PMC Installation Diagram**

### 2.9.1 PMC Bus Speed Selection

When debugging a new PMC module or driver software, or to reduce the CPU load when the PMC module(s) are operating at 66 MHz, it can sometimes be an advantage to reduce the PCI bus clock speed. A switch is provided to force the PCI bus connecting both PMC sites, the dual Ethernet controller and the CPU to operate at 33 MHz. The location and settings of the PMC Bus Speed switch are shown in Figure 2-13.

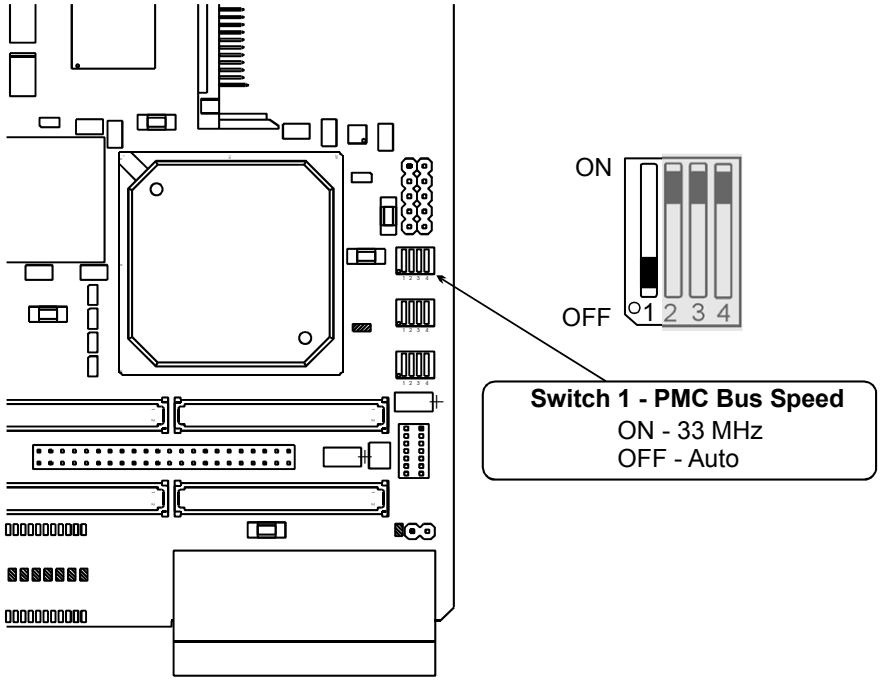



**Figure 2-13 PMC Bus Speed Select Switch**

## 2.10 CompactPCI Operating Mode Selection

This is normally automatic and depends only on what type of slot the board is installed into, as detailed below:

| Slot | Mode |
|---|---|
| System Controller | System Controller |
| Bussed Peripheral | Peripheral or Satellite |
| Non-Bussed Peripheral | Satellite |

**NOTE:** Boards fitted with the PCI 6154 PCI to PCI bridge cannot operate as Peripheral boards. They can only operate as System Controller or Satellite. When installed in a peripheral slot, boards fitted with the PCI 6154 bridge will automatically operate in Satellite mode. For boards fitted with the PCI 6254 bridge, installed in the same slot and required to operate in Satellite mode, this mode must be forced as described below.

A switch is provided to force Satellite mode operation in any slot. The settings are shown in Figure 2-14. In Satellite mode the board cannot communicate on the CompactPCI bus.

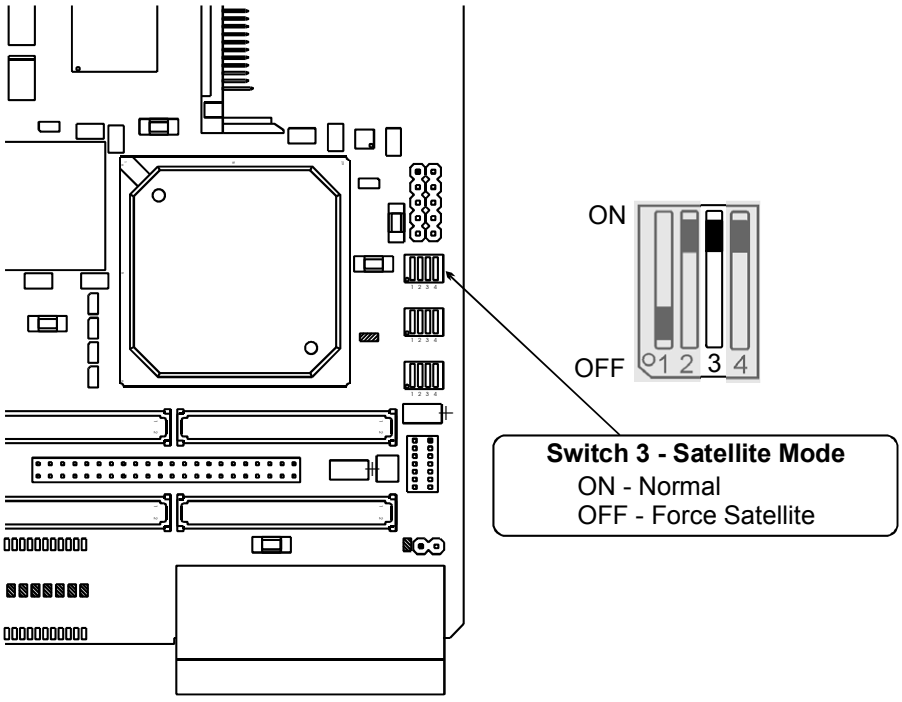



**Figure 2-14 Satellite Mode Switch**

## 2.11 Reset Sources

In addition to the front panel switch described in Section 2.4.8, the board may be reset from several external sources, as described below. Table 2-1 outlines how board and system resets can be achieved using the available jumper options.

| Mode | Board Level Reset Sources | System Reset Sources | Section |
|---|---|---|---|
| System Controller | Front Panel Switch<br>CompactPCI Push Button Reset<br>External Reset | Front Panel Switch<br>CompactPCI Push Button Reset<br>External Reset | 2.4.8<br>2.11.3<br>2.11.2 |
| Peripheral | Front Panel Switch<br>CompactPCI RST# signal<br>External Reset | No option | 2.4.8<br>2.11.1<br>2.11.2 |
| Satellite | Front Panel Switch<br>External Reset (AD PP5/001-3x)<br>CompactPCI RST# signal (optional) | No option | 2.4.8<br>2.11.2<br>2.11.1 |

**Table 2-1 Reset Configuration Options**

**NOTE:** Resets generated by hardware or software which cause a local PCI bus reset will also activate the CompactPCI backplane reset if the board is the System Controller.

### 2.11.1 CompactPCI Reset

The CompactPCI Reset signal is generated by the System Controller and is routed to all bussed Peripheral slots.

If the board is in Satellite mode, it may be preferable for it to ignore the CompactPCI Reset signal.

A switch is provided to facilitate this choice. The settings are shown in Figure 2-15.

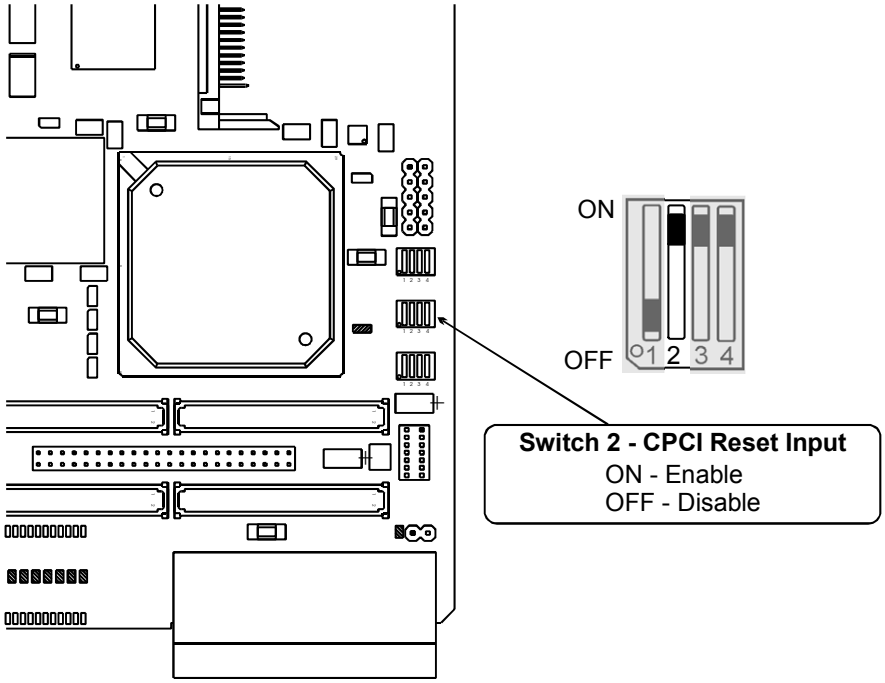



**Figure 2-15 CompactPCI Reset Input Switch**

### 2.11.2 External Reset

When the PP 33x/02x board is used with an AD PP5/001-3x Transition Module, a local (board-level) reset may be generated from a connector on the AD PP5/001-3x (see that board's Technical Reference Manual for details). The action of that connector input is controlled on the PP 33x/02x by the External Reset switch shown in Figure 2-16. This switch has no function when the PP 33x/02x is used without a rear transition module.

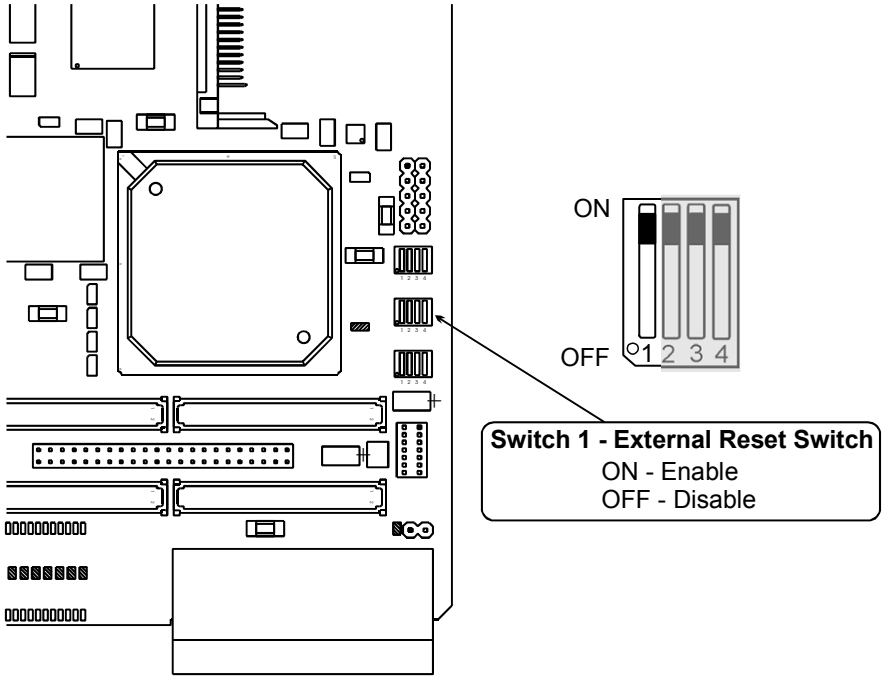



**Figure 2-16 External Reset Switch**

**NOTE:** Unlike the Push Button Reset, this facility is available when the board is installed in any slot.

### 2.11.3 CompactPCI Push Button Reset

The Push Button Reset signal available on the J2 connector (PRST#) will cause a board reset if the board is in the System Controller slot. This input can be driven from an open collector TTL output (or discrete transistor) or normally open switch/relay contacts. To initiate the reset pull this input to 0V. This input is filtered and protected from overshoots/undershoots so no external contact debouncing is required.

This signal is not wired to Peripheral or Satellite slots.

A switch determines whether or not assertion of this signal will reset the board. The settings are shown in Figure 2-17.

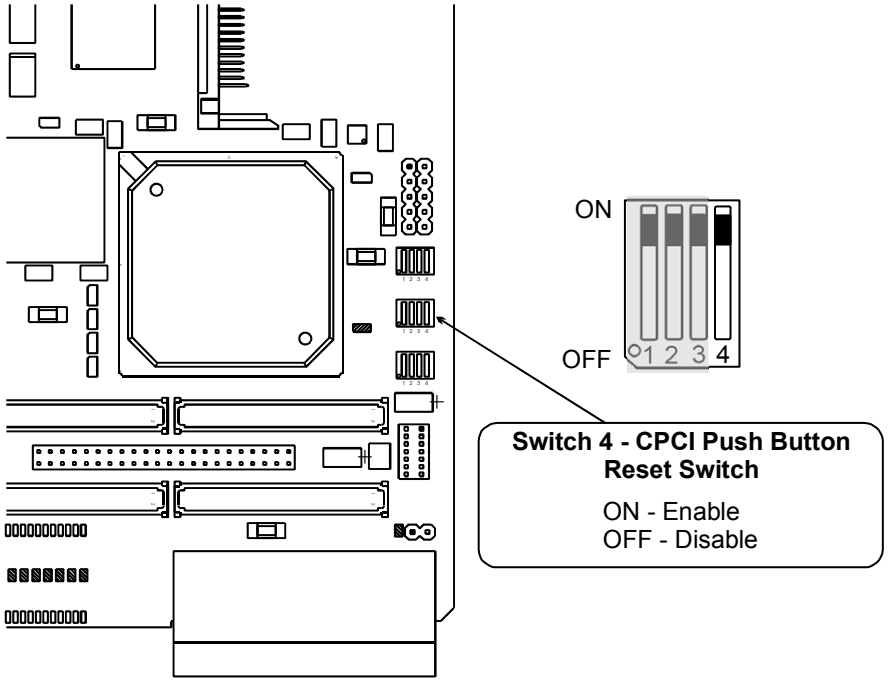



**Figure 2-17 CompactPCI Push Button Reset Switch**

## 2.12 Installation and Power-up

Before the board is installed in a CompactPCI chassis, check the following points:

- The backplane V(I/O) configuration. The PP 33x/02x board supports either 5V or 3.3V V(I/O) automatically. Some CompactPCI backplanes are pre-wired for a particular voltage and others can be configured by the user. For 66 MHz CompactPCI operation the voltage must be 3.3V.

**WARNING:** V(I/O) must be wired on the backplane. If it is not wired the board will hang during POST.

- The Power Supply Unit current capabilities. The board draws current primarily from the +5V and 3.3V rails, and the details are provided in Section A.4.
- Front panel keys. PICMG 2.10 describes a method of keying CompactPCI board front panels to individual chassis slots. Keying pins for this purpose are supplied with the board but are not fitted. The user is referred to PICMG 2.10 for information on using these keys.
- Rear Transition Module Configuration. The AD PP5/001-3x Transition Module is intended for use with the PP 33x/02x board, and has switch options to configure it for rear panel Ethernet or PICMG 2.16. Refer to the AD PP5/001 Technical Reference manual for further details.

The board can be installed as a System Controller in the System slot or as a Peripheral or Satellite board in a Peripheral slot. In Satellite mode the CompactPCI interface is disabled and hence the board will not be recognized by the System Controller. When acting as the System Controller, the board cannot be Hot Swapped.

**NOTE:** The board must be fitted with a PCI 6254 CompactPCI bridge to function as a Peripheral board. Most Rev B and Rev C boards are not fitted with this bridge. Such boards will operate as Satellites if installed in a Peripheral slot.

### 2.12.1 Non Hot Swap Procedure - Installing

If your system requires the use of the EMC spring contact strips provided, fit strips into the slots on the long edges of the front panel.

The board is installed and powered up as follows:

a) Make sure that system power is turned OFF.
b) Slide the board into the designated slot, making sure that the board fits neatly into the runners.
c) Push the board into the card-cage until the J1 ... J5 connectors are firmly located. Use the injector/ejector handles for the final push.
d) Screw the ejector handle retaining bolts into the holes in the chassis.
e) Connect the I/O cables to the connectors on the board's front panel and fix in place with the connectors' retaining screws.
f) If using a Transition Module, install it at the rear of the backplane and connect the I/O cables.
g) Power-up the system. The following sequence of events should then occur:
   - The green "RUN" LED and the yellow "POST" LED on the front panel will light.
   - The yellow "POST" LED will switch OFF.

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

**NOTE:** This sequence of events assumes the PP 33x/02x has Concurrent Technologies standard BIOS firmware and that the board is configured to the factory setting described in Section 2.3.

### 2.12.2 Non Hot Swap Procedure - Removing

To remove the board, shut down the application and operating system software before powering down the system, opening the ejector handles and extracting the board.

### 2.12.3 Hot Swap Procedure - Installing

The board is installed and powered up as follows:

a) Slide the board into the designated slot, making sure that the board fits neatly into the runners.
b) Push the board into the card-cage until the J1 ... J5 connectors are firmly located. Use the injector/ejector handles for the final push.
c) The following sequence of events should then occur:
   - The blue "Hot Swap" LED will flash once.
   - The green "RUN" LED and the yellow "POST" LED on the front panel will light.
   - The yellow "POST" LED will switch OFF.

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

a) Screw the ejector handle retaining bolts into the holes in the chassis
b) Connect the I/O cables to the connectors on the board's front panel and fix in place with the connectors' retaining screws.
c) If using a Transition Module, install it at the rear of the backplane and connect the I/O cables.

### 2.12.4 Hot Swap Procedure - Removing

To remove the board:

a) Open the lower ejector handle and wait for the blue “Hot Swap” LED to switch on. This may take a few seconds. Newer handles require a red button to be pressed in order to open the handle.
b) Open both ejector handles and remove the board.

# 3 SOFTWARE INSTALLATION

In most cases, installing operating system software on the PP 33x/02x board follows the same sequence as installing on a PC. However, there are some additional points to note. The sections below summarize the special actions required for a few common operating systems.

## 3.1 Starting up for the first time

Many operating systems running on the board will want to use the standard Real-Time Clock hardware. To maintain the date and time settings, and several other settings recorded by the PC BIOS, the battery must be fitted. When the board is first powered up, or at the first power-up after changing the battery, carry out the following steps to set up the board.

1) Fit a battery as shown in Section 2.8.
2) Make sure that the Console Mode switch is set to the correct state for the console device which will be used (VGA monitor and keyboard, or serial terminal). Most operating systems which install on the target hardware will require a monitor and keyboard during installation, even if they can subsequently be re-configured to use only a serial terminal. See Section 6.1 for details of how to configure the board for this option.
3) Connect any additional modules and peripherals especially any mass storage devices.
4) Connect the console device and power up the board. Wait for the PC BIOS to sign on and run its memory test.
5) When the test finishes, the BIOS may report a setup or date/time setting error. If this occurs, press the <F2> key as soon as possible after the error is reported, and carry out the following:
6) Set the time and date by using the cursor keys to move aroung the screen and reading the help information in the right hand screen panel.

When the time and date have been set, move the cursor to any other field on the same screen, then press the <F4> key to exit.

Press the ‘y’ key to accept the changes and restart.

The BIOS will then completely restart and re-run its memory test. This time it should complete and begin bootloading. To proceed with software installation, check that all necessary mass storage devices are connected before continuing with one of the sequences below.

# SOFTWARE INSTALLATION

## 3.2 Bootloading from CD-ROM

Operating systems which install on the target hardware will generally install from CD-ROM, or may require both a CD-ROM and floppy disk. Bootloading from floppy disk requires no special steps other than to connect the drive using an appropriate cable. To bootload from CD-ROM, use the following procedure:

a) While the BIOS is running its memory test, press the <ESC> key.
b) Wait for the pop-up boot device menu to be displayed.
c) Select the CD-ROM drive using the cursor keys, then press the <Enter> key.

## 3.3 Installing Microsoft® Windows® Operating Systems

Installing these operating systems on the PP 33x/02x is generally very similar to installing them on a desktop PC. However, Concurrent Technologies also offers a Board Support Package onCD-ROM (part number CD WIN/PC1-L0) which provides installation and configuration information, including appropriate drivers. Please refer to your supplier for further details or to obtain this package.

**NOTE:** For Windows 2000 it may be necessary to reassign the default interrupt allocation for the serial ports, in the BIOS, to prevent a resource conflict being reported. Please refer to Section 10 for more information on the BIOS setup.

## 3.4 Installing RedHat® Linux® 8.0 and 9.0

To install RedHat Linux from CD-ROM, set up the board initially using the steps outlined in Sections 3.1 and 3.2 above, ensuring that all the necessary drives are connected.

The standard distribution of RedHat Linux 8.0 does not currently handle the Intel Pentium M processor, so Concurrent Technologies can provide instructions and additional software to overcome this. In most respects, the installation process is the same as for a standard PC.

Installation of RedHat Linux 9.0 requires additional software in the form of an installation boot disk. Concurrent Technologies can supply an appropriate boot disk and guidance notes for the installation process. Please contact your supplier for further details or to obtain a copy of this package.

# 4 MASS STORAGE INTERFACES

The PP 33x/02x board has two interfaces which can be used to attach mass storage devices:

- Primary EIDE (ATA100) interface is accessible via the CompactPCI J5 connector
- Secondary EIDE (ATA100) interface supporting the on-board Mass Storage option kits

The order in which the PC BIOS firmware tries to bootload from these drives can be changed via the BIOS Setup screen for **Boot**.

## 4.1 EIDE Interfaces

The board supports two EIDE (ATA100) interfaces.

The Primary EIDE interface connects via the CompactPCI J5 connector of the PP 33x/02x board, or through the Transition Module. Up to two EIDE peripherals may be connected to this interface. The BIOS Setup screens for **Main | Primary Master** and **Main | Primary Slave** allow the user to see what is connected to this interface, and to select some characteristics of the drives manually. Normally the PC BIOS firmware will automatically determine the drive characteristics from the drives themselves.

The Secondary EIDE interface connects to the on-board CompactFlash socket and to a 44-pin header for use by the optional Hard Disk or CompactFlash Mass Storage Kits. Up to two EIDE peripherals may be connected to this interface. The on-board CompactFlash socket can be configured as the Secondary Master or the Secondary Slave device as selected by a switch (see Section 2.5). The Hard Disk kit will appear as the Secondary Master device. The CompactFlash modules on the CompactFlash kit will appear as the Secondary Master and Secondary Slave devices.

Note that when using faster EIDE drives the overall cable length from the PP 33x/02x board to the drive furthest from the board must be kept as low as possible, and in any case no more than 18 inches or 450 mm. If this is not practical, it may be necessary to reduce the interface performance using the **UltraDMA Mode** and **Transfer Mode** fields of the BIOS Setup screens indicated above.

To achieve the faster speeds (above ATA33) via the Primary EIDE Interface, it will also be necessary to use the correct (80-way) type of EIDE cable, and to manually select the **User** configuration and **UDMA 4** or **UDMA 5** speeds for the drive using the BIOS Setup screens for **Main|Primary Master** and **Main|Primary Slave** as appropriate. Selection of the fastest speed for the Secondary (on-board) EIDE interface is automatic.

This page is intentionally unused.

# 5 ETHERNET INTERFACES

The PP 33x/02x board is fitted with four independent 1 Gigabit Ethernet interfaces, implemented with an Intel 82546GB controller and two Intel 82541PI controllers. The 82546GB interfaces can connect in different ways, depending primarily on the build configuration of the board. This configuration is indicated by the first digit of the board name suffix. For example, the name PP 330/022-11 indicates a board built for Rear Ethernet. The name PP 330/022-21 indicates the same board but built for backplane networking (PICMG 2.16). The 82541PI interfaces are wired to the front panel Ethernet connectors.

**NOTE:** Earlier revisions of the board were fitted with Intel 82540EM Gigabit Ethernet controllers instead of the 82541PI controllers.

By convention, in this manual the Ethernet interfaces are numbered as 0 and 1 for the 82541PI devices, and 2 and 3 for the dual-channel 82546GB. Operating system drivers may renumber these devices to conform to the driver conventions.

It is possible to boot-load the board through the Ethernet interfaces by setting appropriate BIOS configuration options. See Section 10.3 for more details.

On extended temperature boards some restrictions apply to Ethernet operation at Gigabit speed. Refer to Section 1.5 for more details.

## 5.1 Rear Ethernet Configuration

If the board is built for operation with Rear Ethernet, both of the 82546GB interfaces connect via J3 to a Transition Module which provides the isolating “magnetics” to support standard RJ45 connectors and Category 5 cable.

## 5.2 PICMG 2.16 Configuration

If the board is built for operation with backplane networking (PICMG 2.16), both of the 82546GB interfaces connect via J3, but the isolating “magnetics” are now also built into the PP 33x/02x board itself. This allows the board to connect directly to fabric boards via the J3 sub-plane of a PICMG 2.16 backplane. If a Transition Module is used, it must be wired or jumpered such that it does not connect to the pins on J3, which provide the Ethernet signals. The AD PP5/001-3x Transition Module is fitted with switches to provide this isolation where necessary.

This page is intentionally unused.

# 6 OTHER INTERFACES

Many additional standard interfaces are provided on the PP 33x/02x board. These interfaces consist primarily of those found in a regular desktop or mobile PC, and are outlined below.

## 6.1 Serial Ports

One RS232 serial interface is always available on the PP 33x/02x board, and connects via an in-board 10-way header.

Two serial ports are provided by the AD PP5/001-3x Transition Module.

Table 6-1 summarizes the available serial port options and the location of their connectors.

| Configuration | COM1 | COM2 | COM3 |
|---|---|---|---|
| PP 33x/02x; no Transition Module | PP 33x/02x in-board header | N/A | N/A |
| PP 33x/02x and AD PP5/001-3x | PP 33x/02x in-board header | AD PP5/001-3x | AD PP5/001-3x |

**Table 6-1 Serial Port Numbering**

All the serial ports are implemented in the PC chipset used on the board and Transition Module, using standard 16550-style devices. The two serial ports on the Transition Module, COM2 and COM3, may be configured for speeds up to 115.2kbaud. The onboard serial port, COM1, may also be configured for speeds up to 115.2kbaud.

**NOTE:** On earlier revisions of the board, the onboard serial port could only be configured for speeds up to 38.4kbaud.

### 6.1.1 PC BIOS Serial Console

With some operating systems, or in some applications, it is preferable to use a serial terminal as an operator console device for the board. In other cases, a standard VGA screen and PC keyboard will be required.

The PP 33x/02x does not provide a graphics interface and hence will normally be configured for operation with a serial console. When configured in this mode, the PC BIOS firmware will re-direct its output to a serial port, and similarly will take its input from this port, rather than using the VGA screen and PC keyboard. A board switch selects the console mode, and is shown in Figure 6-1 below.

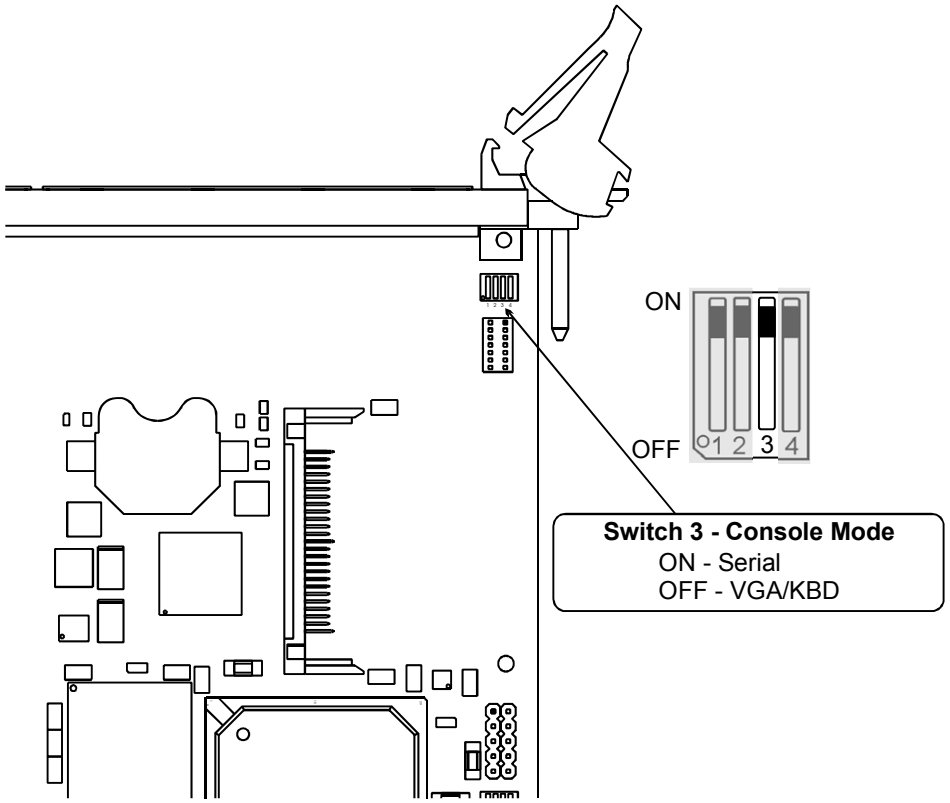



**Figure 6-1 Console Switch**

The serial line speed used for the Serial Console mode may be selected from the BIOS Setup screen for **`Main`** configuration.

### 6.1.2 PC BIOS Serial Console Port

When the PC BIOS is configured to use a serial port for its console, either COM1 or COM2 (on the AD PP5/001-3x Transition Module) can be selected, using the board switch shown in Figure 6-2.

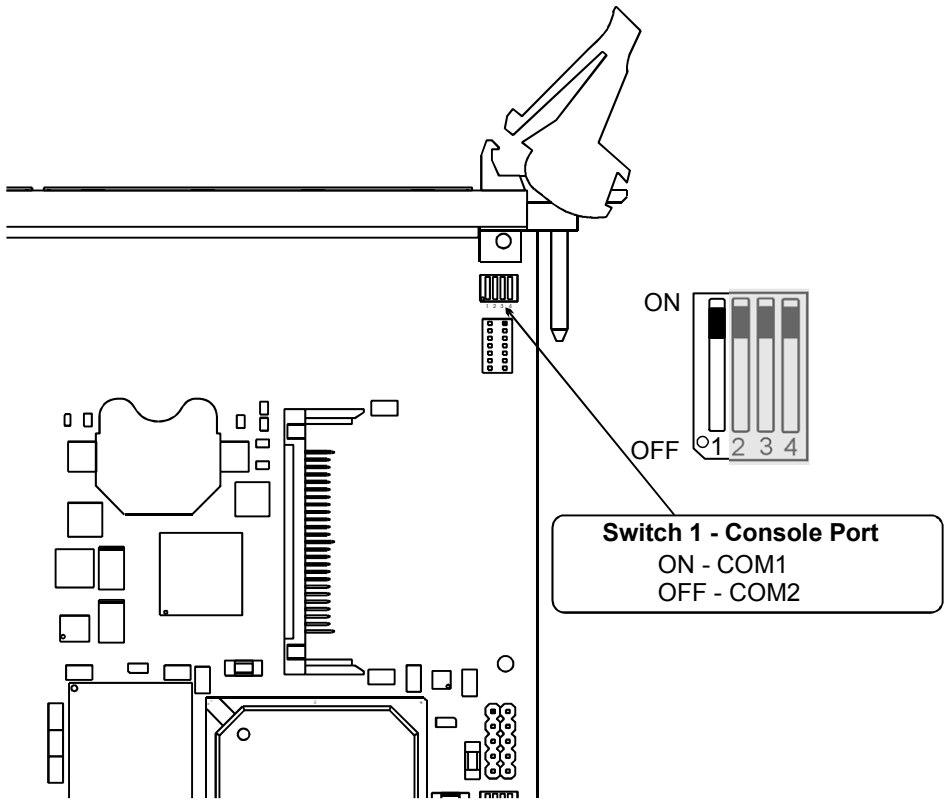



**Figure 6-2 Console Port Switch**

# OTHER INTERFACES

## 6.2 Keyboard and Mouse Ports

The PP 33x/02x supports USB keyboard and mouse via the two front panel USB interfaces. PS/2™ type keyboard and mouse interfaces are not supported.

## 6.3 Graphics (VGA) Controller

The PP 33x/02x does not normally provide an on-board graphics interface.

The graphics controller integrated into the 855GME GMCH is not accessible and is disabled by the PC BIOS firmware.

See also Section 6.1 for information about how to switch the PC BIOS console device between VGA screen and keyboard, and a serial port.

## 6.4 Real-Time Clock

A conventional PC Real-Time Clock is included on this board. This is Year 2000 compliant and can be powered by an on-board Lithium battery when main power to the board is removed. See Section 2.8 for more details of how to fit or replace the battery. The Clock device also provides 242 bytes of CMOS RAM, in which the PC BIOS keeps much of its setup screen data and other information.

## 6.5 Universal Serial Bus (USB)

Three USB interfaces are provided on this board. Two of them are routed to front panel connectors, the third is connected to the CompactPCI J5 connector. All three USB channels support both USB 1.1 and USB 2.0 and can operate at 1.5Mbits/s, 12Mbits/s or 480Mbits/s.

## 6.6 Power On Self Test LED/Speaker

The Power On Self Test (POST) LED is connected to the PC Speaker port. A connection for this port is also made available via the CompactPCI J5 connector. This connection is driven by an open-collector device and can drive a speaker of 32 Ohm or 64 Ohm impedance.

# 7 INTELLIGENT PLATFORM MANAGEMENT INTERFACE

## 7.1 Introduction

The Intelligent Platform Management Interface (IPMI) is an industry-standard environment which allows centralized monitoring and control of a computer system. The features of the IPMI support the management of CompactPCI or other systems containing multiple intelligent modules and other standard features such as sensors for system temperature, fan failure or chassis intrusion. The IPMI specifications define the protocols used by multiple intelligent devices conforming to these specifications to communicate with each other, and in some cases with external systems used for remote monitoring and control. Provision is made in these specifications for non-volatile storage of inventory, control and status information, and to allow the IPMI subsystem to implement other system-specific features.

Although the IPMI specifications can be applied to many different types of system, the remainder of this chapter considers only how they apply to CompactPCI systems containing the PP 33x/02x board. In addition, the IPMI features of this board would normally be used with System Management Software, and the specifications and design of this software are beyond the scope of this manual. For an in-depth discussion of IPMI, the reader is referred to the original IPMI specifications. At the time of writing, these specifications are available from the World Wide Web, at the address:

http://developer.intel.com/design/servers/ipmi/index.htm

In a practical implementation, the features of an IPMI may be used by System Management Software to implement a complete control and monitoring application. The specifications and design of this software are beyond the scope of this manual. The System Management Software will normally be run on the CompactPCI system controller board and in this role the PP 33x/02x board's local IPMI subsystem will be termed the Baseboard Management Controller (BMC). In this role the board acts as the interface between the System Management Software and the CompactPCI system chassis. The BIOS Setup screen for **Advanced** allows the user to force the board to act either as a BMC or as a Slave Management Controller (SMC) in any slot.

The remainder of this chapter outlines the functions provided by this board via IPMI, and details some ways in which these features can be used. For a more complete description of the IPMI protocols and implementation on Concurrent Technologies' boards, refer to the document: "Intelligent Platform Management Interface for Concurrent Technologies Boards", order code 555 0042.

## 7.2 IPMI Compatibility

This board implements a version of IPMI compatible with revision 1.5 of the IPMI specifications. It includes the mandatory elements of the PCI Industrial Computer Manufacturer's Group (PICMG) specification 2.9.

The IPMI facilities supported by the PP 33x/02x board are implemented using a microcontroller and its resident firmware with non-volatile operational data stored in EEPROM. A total of 8 Kbytes of non-volatile storage is provided for inventory, sensor and event data recording. This implementation provides a hardware interface which conforms to the Server Management Interface Controller (SMIC) type as defined in the IPMI specifications.

## 7.3 IPMI Overview

The PP 33x/02x board includes hardware and firmware which implements an IPMI as a separate resource to the main Intel Pentium M processor. The processor communicates with the IPMI subsystem through a standardized hardware interface using multi-byte message sequences, transferring one byte at a time using a handshaking protocol. Instructions can be sent to the IPMI subsystem with a command and response message pair. The IPMI subsystem may in turn communicate with additional hardware in the chassis using similar message sequences transferred over two Intelligent Platform Management Busses (IPMB). The IPMB 0 and IPMB 1 busses each provide a 2-wire interface based on the Philips® I2C protocol, and in the CompactPCI chassis connects to all boards on the CompactPCI backplane via the J1 and J2 connectors.

The IPMB 0 bus and IPMB 1 bus can connect to additional intelligent or non-intelligent devices which may act as sensors or simply data sources and sinks. These devices may include fan or power supply monitoring hardware, temperature sensors or perhaps just non-volatile memory. The IPMI specifications define data structures for these sensors which are stored in non-volatile memory, and describe both control and run-time status (event) information that can be retrieved by System Management Software.

As the IPMI subsystem on the PP 33x/02x board is separated from the main processor, many of its features can operate when main power is removed, provided that power is supplied to the board through the IPMB_PWR pin of the J1 connector. This allows the board to be interrogated via the IPMB 0 bus even in a power-down state, which may be useful when the board is operating in Satellite mode.

The programming interface to the IPMI subsystem is via the System Management Interface Controller (SMIC). This is a set of I/O registers accessed using a polled handshaking protocol. Example software for driving this interface is provided in Section 7.6.1. When the board is acting as an SMC, software running on the local processor may only access local IPMI resources through the SMIC interface. When the board is acting as the BMC, software running on the local processor may access both on-board IPMI resources and those elsewhere on the IPMB.

The implementation of IPMI on this board provides the following functions.

### 7.3.1 Message Passing

The flow of information in an IPMI compliant system is achieved by using messages. A transaction consists of a request and a response message pair. The interface between the System Management Software and the Baseboard Management Controller uses simple messages which contain enough information to allow a response to be generated. The “Get Message” and “Send Message” commands are used to pass information to the IPMB and embed simple messages with channel routing information.

### 7.3.2 Events

Events can be generated whenever a system failure is detected. These events are stored in the System Event Log and can be retrieved by the System Management Software which can process the events and determine what, if any, corrective action can be taken.

### 7.3.3 System Event Log

Events are stored in the System Event Log which is held in non-volatile memory. The System Management Software can access the System Event Log and by analyzing the events may be able to determine the sequence of events that caused the system failure.

### 7.3.4 Sensors

The PP 33x/02x board is capable of monitoring the temperature of the board and the processor chip, the levels of the main power supply voltage rails, the status of a system fan and the board's Geographic Address (effectively its CompactPCI bus slot number). These sensors can be configured to generate events when, for example, the temperature of the processor chip exceeds a previously configured value. Sensor Data Records are used to contain information pertaining to sensors.

### 7.3.5 Sensor Data Records

Sensor Data Records (SDRs) are used to describe the configuration and characteristics of sensors and can also be used to describe the operating characteristics of the Intelligent Platform Management Device. These Sensor Data Records are stored in a repository held in non-volatile memory. The PP 33x/02x board is supplied with preconfigured SDRs describing its onboard sensors.

### 7.3.6 Field Replaceable Unit Inventory Data

The Field Replacement Unit (FRU) Inventory Data is stored in non-volatile memory and allows the System Management Software to determine the identity of Intelligent Platform Management devices in the system. Typically this data comprises the manufacturer's name, board name, part number, serial number, etc.

### 7.3.7 Watchdog

The Watchdog function allows the System Management Software to protect the system from errant programs. For example, a critical program taking 3 seconds to complete may set the watchdog to expire in 5 seconds. If the watchdog expires then the system management software can take the appropriate corrective action.

## 7.4 Supported Commands

The PP 33x/02x board supports a subset of the command messages as defined in the IPMI specification. All commands, unless explicitly stated, expect a response which is generally retrieved through the SMIC interface. Commands that are not supported receive a “Invalid Command” response.

Table 7-1 list the commands supported by this board. Many of these commands are described in the IPMI specifications and no further details are provided here. The following sub-sections indicate board specific characteristics of these functions. For a complete description of the commands, refer to the documents indicated in Section 7.1.

| | | NetFn | CMD |
|---|---|---|---|
| **IPM Device "Global" Commands** | Get Device ID | 06h | 01h |
| | Broadcast Get Device ID | 06h | 01h |
| | Cold Reset | 06h | 02h |
| | Warm Reset | 06h | 03h |
| | Get Self Test Results | 06h | 04h |
| **BMC Watchdog Timer Commands** | Reset Watchdog Timer | 06h | 22h |
| | Set Watchdog Timer | 06h | 24h |
| | Get Watchdog Timer | 06h | 25h |
| **BMC Device and Messaging Commands** | Set BMC Global Enables | 06h | 2Eh |
| | Get BMC Global Enables | 06h | 2Fh |
| | Clear Message Flags | 06h | 30h |
| | Get Message Flags | 06h | 31h |
| | Get Message | 06h | 33h |
| | Send Message | 06h | 34h |
| | Get Channel Info | 06h | 42h |
| | Master Write-Read | 06h | 52h |
| **Chassis Device Commands** | Get Chassis Capabilities | 00h | 00h |
| **Event Commands** | Set Event Receiver | 04h | 00h |
| | Get Event Receiver | 04h | 01h |
| | Platform Event Message | 04h | 02h |
| **Sensor Device Commands** | Set Sensor Threshold | 04h | 26h |
| | Get Sensor Threshold | 04h | 27h |
| | Get Sensor Reading | 04h | 2Dh |
| **FRU Device Commands** | Get FRU Inventory Area Info | 0Ah | 10h |
| | Read FRU Data | 0Ah | 11h |
| | Write FRU Data | 0Ah | 12h |
| **SDR Device Commands** | Get SDR Repository Info | 0Ah | 20h |
| | Get SDR Repository Allocation Info | 0Ah | 21h |
| | Reserve SDR Repository | 0Ah | 22h |
| | Get SDR | 0Ah | 23h |
| | Add SDR | 0Ah | 24h |
| | Partial Add SDR | 0Ah | 25h |
| | Clear SDR Repository | 0Ah | 27h |
| | Get SDR Repository Time | 0Ah | 28h |
| | Set SDR Repository Time | 0Ah | 29h |
| **SEL Device Commands** | Get SEL Info | 0Ah | 40h |
| | Get SEL Allocation Info | 0Ah | 41h |
| | Reserve SEL | 0Ah | 42h |
| | Get SEL Entry | 0Ah | 43h |
| | Add SEL Entry | 0Ah | 44h |
| | Delete SEL Entry | 0Ah | 46h |
| | Clear SEL | 0Ah | 47h |
| | Get SEL Time | 0Ah | 48h |
| | Set SEL Time | 0Ah | 49h |

**Table 7-1 Supported IPMI Commands**

### 7.4.1 Get Self Test Results Command

On this board, no self test results are available via this command. The PC BIOS runs a Power On Self Test (POST) sequence, but the results of these tests are reported only via on-screen messages and flashes of the POST LED.

The IPMI command always returns code 56h (meaning not supported).

### 7.4.2 Watchdog Commands

The basic timing function of the watchdog is controlled by a 16-bit timer with 100ms ticks giving a watchdog timing period between 0 and 6553.5 seconds. The watchdog can optionally be programmed to generate either NMI or SMI interrupts at a fixed interval (specified in seconds) before the watchdog expires. This is termed a pre-interrupt. The watchdog when it expires can be programmed to perform no action, reset the board and either power-off or power cycle the board. The watchdog can also generate an internal event which is stored in the System Event Log.

**NOTE:** The watchdog is disabled following a power on or a reset

Example source code for controlling the Watchdog timer is provided in Section 7.6.2.

### 7.4.3 SEL and SDR Commands

The SEL and SDR Repository Timestamp counters are the same on this board. This means that operations affecting one of these counters also affect the other.

### 7.4.4 Sensor Commands

The list of sensors fitted to this board, together with the sensor number and type information found in their SDRs, is provided in Table 7-2. These sensors are scanned by the IPMI subsystem at a rate of approximately 10 Hz. The values returned by the Get Sensor Reading command are the values obtained in the most recent scan.

The IPMI subsystem will also include SDRs for IPMB Management Controller Locator and IPMB Management Controller Confirmation. These records provide status information indicating the capability of the board depending on its position in the CompactPCI chassis. They are used primarily by other IPMB devices which want to retrieve information about the other devices on the IPMB bus.

| Sensor Function | Sensor Number | Sensor Type | Event Type |
|---|---|---|---|
| Board Temperature | 00h | 01h | 01h |
| CPU Temperature | 01h | 01h | 01h |
| Fan Monitor | 80h | 0Ah | 03h |
| Voltage, +12V | 10h | 02h | 01h |
| Voltage, +5V | 11h | 02h | 01h |
| Voltage, +3.3V | 12h | 02h | 01h |
| Voltage, +2.5V | 13h | 02h | 01h |
| Voltage, +1.8V | 14h | 02h | 01h |
| Voltage, CPU Core | 15h | 02h | 01h |
| Voltage, VCCP | 16h | 02h | 01h |
| Voltage, VTT | 17h | 02h | 01h |
| cPCI Slot Number | 30h | C0h | 6Fh |
| cPCI BDSEL Signal | 31h | C1h | 03h |
| cPCI SYSEN Signal | 32h | C1h | 03h |
| cPCI PRESENT Signal | 33h | C1h | 03h |
| Hot Swap Ctrl Power Good | 34h | 02h | 06h |
| Hot Swap Ctrl Power Fail | 35h | 02h | 06h |
| Ejector Handle Status | 3Ah | C1h | 03h |
| Power Regulators Status | 3Bh | 08h | 06h |

**Table 7-2 Sensor ID Codes**

#### 7.4.4.1 Board Temperature Sensors

A National Semiconductor LM86 device measures the local board and CPU temperature with a resolution of 0.125°C. The device has a quoted accuracy of +/- 1°C over a range of +60°C to +100°C, and of +/- 3°C outside this range.

#### 7.4.4.2 Fan Monitor

Fan failure is detected by a single change in the state of the fan monitor input on the J5 connector. The signal state which indicates failure is set in the Sensor Data Record (SDR) for this sensor. The Assertion Event Mask / Lower Threshold Reading Mask and Deassertion Event Mask / Upper Threshold Reading Mask fields are used to set these parameters. The default settings for these fields select an active low fan failure indication. To select an active high fan failure indication, the pre-defined SDR must be deleted and replaced by one which sets these fields to 0002h.

#### 7.4.4.3 Board Voltage Sensors

The microcontroller measures various on-board voltages, namely: +12V, +5V, +3.3V, 2.5V, 1.8V, VTT (memory bus terminations), VCPU (processor core) and VCCP (front side bus terminations). The sensor reading is a raw value and must be converted to Volts by application software. The formulas are (where x is the raw sensor reading):

| | |
|---|---|
| +12V | = 0.0731x |
| +5V | = 0.0234x |
| +3.3V | = 0.0188x |
| +2.5V | = 0.0171x |
| +1.8V | = 0.00947x |
| CPU CORE | = 0.00947x |
| VCCP | = 0.00947x |
| VTT | = 0.00947x |

These conversion factors are also in the SDR associated with each sensor so application software can read them from there rather than hardcode them.

#### 7.4.4.4 CompactPCI Slot Number

This sensor reading indicates board slot number which is read from geographic address pins on the CPCI backplane. The possible values are from 0 to 31.

#### 7.4.4.5 CompactPCI BDSEL Signal

This sensor reading indicates status of the BDSEL# pin on the CPCI backplane. The board is fully seated if the value is 1. If the value is 0, the board is not fully seated.

#### 7.4.4.6 CompactPCI SYSEN Signal

This sensor reading indicates status of the SYSEN# pin on the CPCI backplane. The board is in the system controller slot if the value is 1. If the value is 0, the board is not in the system controller slot.

#### 7.4.4.7 CompactPCI PRESENT Signal

This sensor reading indicates status of the PCI_PRESENT# pin on the CPCI backplane. The PCI bus is present at this slot if the value is 1. If the value is 0, the PCI bus is not present at this slot. This sensor is provided mainly for use with PICMG 2.16 backplanes.

#### 7.4.4.8 Hot Swap Control Power Good

This sensor reading indicates status of power supply voltages from the backplane. If the value is 0, the power supply voltages are within specification. If the value is 1, one or more voltages are out of specification.

#### 7.4.4.9 Hot Swap Control Power Fail

This sensor reading indicates status of power supply currents from the backplane. If the value is 0, the power supply currents are good. If the value is 1, an overcurrent has occurred.

#### 7.4.4.10 Ejector Handle

This sensor reading indicates the status of the ejector handle switch. If the value is 0, the handle is closed. If the value is 1, the handle is open.

#### 7.4.4.11 Power Regulators Status

This sensor indicates if the power regulators are working correctly. If they are, the sensor returns 1. Otherwise, if there is a problem, the sensor returns 0.

## 7.5 FRU Inventory Data

The FRU Inventory Area shares the non-volatile memory with the repository of Sensor Data Records and the System Event Log and contains information about the board, including, for example, the manufacturer's name, part number and serial number.

The FRU Inventory Area comprises, at most, six information areas. Each area, if used, is always a multiple of 8 bytes in length. The Common Header Area is always included and the PP 33x/02x board also includes the Board Area. The Internal Use, Chassis Information, Product Information and Multi-Record areas are currently not used.

### 7.5.1 Common Header Area

The Common Header Area is always included in the FRU Inventory Area and is used to specify the location of the other areas in the FRU Inventory Area. The offset of any area in the FRU Inventory Area must be a multiple of 8h. The Common Header Area always has an offset of 0000h and is arranged as shown in Table 7-3.

| FRU Inventory Area Offset | Value | Meaning |
|---|---|---|
| 0000h | 01h | Common Header Format Version |
| 0001h | 00h | Offset to Internal Use Area |
| 0002h | 00h | Offset to Chassis Information Area |
| 0003h | 01h | Offset to Board Area |
| 0004h | 00h | Offset to Product Information Area |
| 0005h | 00h | Offset to Multi-Record Area |
| 0006h | 00h | Reserved |
| 0007h | FEh | Common Header Area Checksum |

**Table 7-3 FRU Inventory Area : Common Header Area Data**

### 7.5.2 Board Area

The Board Area, in the FRU Inventory, contains information about the PP 33x/02x board. Its offset in the FRU Inventory Area is calculated by multiplying its offset value in the Common Header Area by 8. The Board Area is arranged as shown in Table 7-4.

| Board Area Offset | Value | Comments |
|---|---|---|
| 0000h | 01h | Board Area format version |
| 0001h | 09h | Board Area length |
| 0002h | 19h | Language code |
| 0003h | xxxxxxh | Manufacturing date and time |
| 0006h | D7h | ASCII+LATIN 1 character set and 23 characters |
| 0007h | “Concurrent Technologies” | Manufacturer’s name |
| 001Eh | CAh | ASCII+LATIN 1 character set and 10 characters |
| 001Fh | “PP 33x/02x” | Board’s name |
| 0029h | CAh | ASCII+LATIN 1 character set and 10 characters |
| 002Ah | “M0001/999” | Typical serial number |
| 0034h | CBh | ASCII+LATIN 1 character set and 11 characters |
| 0035h | “760-6018-xx” | Board’s part number |
| 0040h | C0h | No FRU file ID record is define |
| 0041h | C1h | No more records |
| 0042h | 00h | Reserved |
| 0043h | 00h | Reserved |
| 0044h | 00h | Reserved |
| 0045h | 00h | Reserved |
| 0046h | 00h | Reserved |
| 0047h | xxh | Board Area checksum |

**Table 7-4 FRU Inventory Area : Board Area Data**

### 7.5.3 Board Area Format Version

This field defines the format of the Board Area and is always set to 01h.

#### 7.5.3.1 Board Area Length

This size of the Board Area in the FRU Inventory Area is computed by multiplying this field by 8.

#### 7.5.3.2 Language Code

The language code is always set to 19h (English).

#### 7.5.3.3 Manufacturing Date and Time

This is defined as the number of minutes since 00:00 hours on the 1st January 1996.

#### 7.5.3.4 Manufacturer’s Name

This is a string of characters and is set to “Concurrent Technologies”. It is not zero terminated.

#### 7.5.3.5 Board’s Name

This is a string of characters representing the board’s name and is set to “PP 33x/02x”. It is not zero terminated.

#### 7.5.3.6 Board’s Port Number

This is a string of characters representing the board’s internal part number. It is not zero terminated.

#### 7.5.3.7 Serial Number

This is a string of characters matching the serial number imprinted on the board. It is not zero terminated. An example of a serial number is “M3144/003”.

#### 7.5.3.8 FRU File ID

This field is used to store manufacturing information but is currently unused.

## 7.6 Programming Examples

### 7.6.1 Using the SMIC Interface

The PP 33x/02x board utilizes the Server Management Interface Controller (SMIC) as its interface with the I/O ports at the standard addresses (i.e. 0CA9h, 0CAAh and 0CABh). The IPMI specification has a complete description of the SMIC interface. The following C program fragment reads and writes IPMI messages using the SMIC interface:

```
 /* Addresses of SMIC registers */

#define SMIC_DATA       0x0CA9  /* Data Register */
#define SMIC_CONTROL    0x0CAA  /* Control & Status Register */
#define SMIC_FLAGS      0x0CAB  /* Flag Register */

/* SMS Transfer Stream Control Codes */

#define CC_SMS_GET_STATUS   0x40
#define CC_SMS_WR_START     0x41
#define CC_SMS_WR_NEXT      0x42
#define CC_SMS_WR_END       0x43
#define CC_SMS_RD_START     0x44
#define CC_SMS_RD_NEXT      0x45
#define CC_SMS_RD_END       0x46

/* SMS Transfer Stream Status Codes */

#define SC_SMS_RDY          0xC0
#define SC_SMS_WR_START     0xC1
#define SC_SMS_WR_NEXT      0xC2
#define SC_SMS_WR_END       0xC3
#define SC_SMS_RD_START     0xC4
#define SC_SMS_RD_NEXT      0xC5
#define SC_SMS_RD_END       0xC6

/* Masks for SMIC Flags Registers Bits */

#define FLAG_RX_DATA_RDY    0x80
#define FLAG_TX_DATA_RDY    0x40
#define FLAG_SMI            0x10
#define FLAG_EVT_ATN        0x08
#define FLAG_SMS_ATN        0x04
#define FLAG_BUSY           0x01

#define FLAGS_BUSY          0x01
#define FLAGS_TX_DATA_READY 0x40
#define FLAGS_RX_DATA_READY 0x80

/* macros */

#define bReadSmicFlags  (inb (SMIC_FLAGS))
#define bReadSmicStatus (inb (SMIC_CONTROL))
#define bReadSmicData   (inb (SMIC_DATA))

#define vWriteSmicControl(data)     outb(SMIC_CONTROL, data)
#define vWriteSmicData(data)        outb(SMIC_DATA, data)
#define vWriteSmicFlags(data)       outb(SMIC_FLAGS, data)
```

```
/*****************************************************************************
 *
 * vBmcSmicSmsMessageWrite
 *
 * This function writes a SMS (System Managment Software) messages
 * to the IPMI using the standard BMC-SMIC interface.
 *
 * Returns: N/A
 *
 */
void vBmcSmicSmsMessageWrite
(
    const unsigned char *pbMessage, /* request */
    unsigned char bLength           /* request length */
)
{
    unsigned char bMessageStage;


    while (((bReadSmicFlags) & FLAGS_BUSY) == FLAGS_BUSY)
        ; /* do nothing ... */

    vWriteSmicControl (CC_SMS_WR_START);
    vWriteSmicData (*pbMessage++);
    vWriteSmicFlags (FLAGS_BUSY);
    bLength--;

    do
    {
        while (((bReadSmicFlags) & FLAGS_TX_DATA_READY) != FLAGS_TX_DATA_READY)
            ; /* do nothing ... */

        while (((bReadSmicFlags) & FLAGS_BUSY) == FLAGS_BUSY)
            ; /* do nothing ... */

        vWriteSmicControl ((bLength > 1) ? CC_SMS_WR_NEXT : CC_SMS_WR_END);
        vWriteSmicData (*pbMessage++);
        vWriteSmicFlags (FLAGS_BUSY);
        bLength--;

        while (((bReadSmicFlags) & FLAGS_BUSY) == FLAGS_BUSY)
            ; /* do nothing ... */
    }
    while (bLength > 0);

}
/*****************************************************************************
 *
 * vBmcSmicSmsMessageRead
 *
 * This function reads a SMS (System Managment Software) message
 * from the IPMI using the standard BMC-SMIC interface.
 *
 * Returns: N/A
 *
```

```
 */
void vBmcSmicSmsMessageRead
(
    unsigned char *pbMessage,       /* received response */
    unsigned char *bMessageLength   /* response length */
)
{
    unsigned char bReadControl;
    unsigned char bReadStatus;
    unsigned char bReadData;

    *bMessageLength = 0;

    bReadControl = CC_SMS_RD_START;
    do
    {
        while (((bReadSmicFlags) & FLAGS_RX_DATA_READY) != FLAGS_RX_DATA_READY)
            ; /* do nothing ... */

        while (((bReadSmicFlags) & FLAGS_BUSY) == FLAGS_BUSY)
            ; /* do nothing ... */

        vWriteSmicControl (bReadControl);
        vWriteSmicFlags (FLAGS_BUSY);
        while (((bReadSmicFlags) & FLAGS_BUSY) == FLAGS_BUSY)
            ; /* do nothing ... */

        bReadStatus = bReadSmicStatus;
        bReadData   = bReadSmicData;

        switch (bReadStatus)
        {
            case SC_SMS_RD_START :
                *pbMessage++ = bReadData;
                (*bMessageLength)++;
                bReadControl = CC_SMS_RD_NEXT;
                break;
            case SC_SMS_RD_NEXT  :
                *pbMessage++ = bReadData;
                (*bMessageLength)++;
                break;
            case SC_SMS_RD_END   :
                *pbMessage++ = bReadData;
                (*bMessageLength)++;
                bReadControl = CC_SMS_RD_END;
                break;
            default :
                break;
        }
    }
    while (bReadStatus != SC_SMS_RDY);

    return;
}
```

### 7.6.2 Using the Watchdog Timer

The IPMI provides a standardized watchdog facility which is fully described in the IPMI specification. The following C program fragment sets and resets the watchdog facility:

```
/* network function codes */

#define NFC_APP_REQUEST         0x06
#define NFC_APP_RESPONSE        0x07

/* watchdog commands */

#define CMD_WATCHDOG_RESET  0x22    /* Reset Watchdog Timer */
#define CMD_WATCHDOG_SET    0x24    /* Set Watchdog Timer */
#define CMD_WATCHDOG_GET    0x25    /* Get Watchdog Timer */

/* definition of watchdog operational constants */

#define USAGE_BIOS_FRB2     1
#define USAGE_BIOS_POST     2
#define USAGE_OS_LOAD       3
#define USAGE_SMS_OS        4
#define USAGE_OEM           5

#define FLAG_BIOS_FRB2      (1 << (USAGE_BIOS_FRB2))
#define FLAG_BIOS_POST      (1 << (USAGE_BIOS_POST))
#define FLAG_OS_LOAD        (1 << (USAGE_OS_LOAD))
#define FLAG_SMS_OS         (1 << (USAGE_SMS_OS))
#define FLAG_OEM            (1 << (USAGE_OEM))

#define PRE_TO_INT_NONE         0
#define PRE_TO_INT_SMI          1
#define PRE_TO_INT_NMI          2
#define PRE_TO_INT_MESSAGE      3

#define TO_ACTION_NONE          0
#define TO_ACTION_RESET         1
#define TO_ACTION_POWER_DOWN    2
#define TO_ACTION_POWER_CYCLE   3

/* Completion codes */

#define COMPLETION_OK           0       /* Completion code OK */

/* error codes */

#define E_OK            0           /* OK */
#define E_COMPLETION    0x400       /* wrong completion code */

/* forward declarations */

void vBmcSmicSmsMessageWrite (const unsigned char *pbMessage,
                              unsigned char bLength);
void vBmcSmicSmsMessageRead (unsigned char *pbMessage,
                              unsigned char *bMessageLength);
```

```
/*****************************************************************************
 *
 * wSetWatchdog
 *
 * This function defines operation of the IPMI watchdog which is initiated
 * by the Reset Watchdog Command issued the the wResetWatchdog function.
 *
 * RETURNS: E_OK if it is OK, or error code
 *
 */
unsigned short int wSetWatchdog
(
    unsigned char bDontLog,                 /* FALSE to log event */
    unsigned short int wTimeoutInterval,    /* multiples of 100ms */
    unsigned char bTimeoutAction,
    unsigned char bPreTimeoutInterval,      /* multiples of 1s */
    unsigned char bPreTimeoutInterrupt,
    unsigned char bTimerUse,
    unsigned char bTimerUseClearFlags
)
{
    unsigned char abRequest  [10];
    unsigned char abResponse [10];
    unsigned char bLength;
    unsigned short int wStatus = E_OK;

    abRequest [0] = NFC_APP_REQUEST << 2;
    abRequest [1] = CMD_WATCHDOG_SET;       /* command */
    abRequest [2] = ((bDontLog) ? 0x80 : 0) | bTimerUse & 0x07;
    abRequest [3] = ((bPreTimeoutInterrupt & 0x07) << 4)
                    | (bTimeoutAction & 0x07);
    abRequest [4] = bPreTimeoutInterval;
    abRequest [5] = bTimerUseClearFlags;
    abRequest [6] = (unsigned char ) (wTimeoutInterval & 0x00FF);
    abRequest [7] = (unsigned char ) (wTimeoutInterval >> 8);

    vBmcSmicSmsMessageWrite (abRequest, 8);
    vBmcSmicSmsMessageRead (abResponse, &bLength);

    if (abResponse [2] != COMPLETION_OK)
        wStatus = E_COMPLETION;
}

/*****************************************************************************
 *
 * wResetWatchdog
 *
 * This function starts / restarts the IPMI watchdog.
 *
 * RETURNS: E_OK if it is OK, or error code
 *
 */
unsigned short int wResetWatchdog (void)
{
    unsigned char abRequest  [10];
    unsigned char abResponse [10];
```

```
    unsigned char bLength;
    unsigned short int wStatus = E_OK;

    abRequest [0] = NFC_APP_REQUEST << 2;
    abRequest [1] = CMD_WATCHDOG_RESET; /* command */

    vBmcSmicSmsMessageWrite (abRequest, 2);
    vBmcSmicSmsMessageRead (abResponse, &bLength);

    if (abResponse [2] != COMPLETION_OK)
        wStatus = E_COMPLETION;
        return wStatus;
}
```

The following example shows how to set the watchdog to power cycle if the watchdog is not restarted within 20 seconds, generate an NMI if there is less than 10 seconds before the watchdog expires, define its use as an “operating system load” watchdog, clear any OEM expiration flags, log the watchdog failure in the event log:

```
wSetWatchdog (FALSE, 200, TO_ACTION_POWER_CYCLE,
                      10, PRE_TO_INT_NMI,
                      USAGE_OS_LOAD, FLAG_OEM);
```

The **wSetWatchdog** function does not start the watchdog facility and the **wResetWatchdog** function must be used. This function simply restarts from the internal watchdog timer to the value specified in the **vSetWatchdog** function. The **wResetWatchdog** function is used thus:

```
wResetWatchdog ();
```

The **vSetWatchdog** function is used to disable the watchdog facilities and is used thus:

```
wSetWatchdog (FALSE, 0, 0, 0, 0, 0, 0);
```

### 7.6.3 Reading Sensors

The IPMI specification supports many commands to manage and interrogate sensors. The following program fragment illustrates how the current reading of a sensor can be obtained.

```
/* network function codes */

#define NFC_SENSOR_EVENT_RQ 0x04

/* commands */

#define CMD_GET_SENS_RD     0x2D    /* Get sensor reading */

/* Completion codes */

#define COMPLETION_OK       0       /* Completion code OK */

/* error codes */

#define E_OK            0           /* OK */
#define E_COMPLETION    0x400       /* wrong completion code */

/* forward declarations */

void vBmcSmicSmsMessageWrite (const unsigned char *pbMessage,
                              unsigned char bLength);
void vBmcSmicSmsMessageRead (unsigned char  *pbMessage,
                              unsigned char *bMessageLength);



/* response data structure provided to wGetSensorReadingCmd()  */

struct SENSOR_READING
{
    unsigned char bData;
    unsigned char bStatus;
};


/*****************************************************************************
 *
 * wGetSensorReadingCmd
 *
 * This function gets current sensor reading.
 *
 * RETURNS: E_OK if it is OK, or error code
 *
 */
unsigned short int wGetSensorReadingCmd
(
    unsigned char bSensorId,
    struct SENSOR_READING *psSensorReading
)
{
    unsigned char bLength;
```

```
    unsigned char abRequest  [10];
    unsigned char abResponse [10];
    unsigned short int wStatus = E_OK;

    /* Send Request */

    abRequest [0] = NFC_SENSOR_EVENT_RQ << 2;
    abRequest [1] = CMD_GET_SENS_RD;        /* command */
    abRequest [2] = bSensorId;

    vBmcSmicSmsMessageWrite (abRequest, 3);
    vBmcSmicSmsMessageRead (abResponse, &bLength);

    if (abResponse [2] != COMPLETION_OK)
        wStatus = E_COMPLETION;
    else
    {
        psSensorReading->bData   = abResponse [3];
        psSensorReading->bStatus = abResponse [4];
    }

    return wStatus;
}
```

The following example shows how to read the CPU temperature sensor and voltage sensors:

```
/* read CPU temperature, sensor ID = 01h */

wStatus = wGetSensorReadingCmd (0x01, &sSensorReading);
if (wStatus == E_OK)
    printf("CPU temperature is %dC\n", sSensorReading.bData);

/* read +12V Power Supply Voltage, sensor ID = 10h */

wStatus = wGetSensorReadingCmd (0x10, &sSensorReading);
if (wStatus == E_OK)
    printf ("Power Supply +12V = %fV\n", sSensorReading.bData * 0.0731);

/* read +5V Power Supply Voltage, sensor ID = 11h */

wStatus = wGetSensorReadingCmd (0x11, &sSensorReading);
if (wStatus == E_OK)
    printf ("Power Supply +5V = %fV\n", sSensorReading.bData * 0.0234);

/* read +3.3V Power Supply Voltage, sensor ID = 12h */

wStatus = wGetSensorReadingCmd (0x12, &sSensorReading);
if (wStatus == E_OK)
    printf ("Power Supply +3.3V = %fV\n",
            sSensorReading.bData * 0.0188);

/* read CPU Power Supply Voltage, sensor ID = 15h */

wStatus = wGetSensorReadingCmd (0x15, &sSensorReading);
if (wStatus == E_OK)
    printf ("Power Supply CPU = %fV\n", sSensorReading.bData * 0.00947);
```

This page is intentionally unused.

# 8 FLASH EPROM AND DRAM

## 8.1 Flash EPROM

The PP 33x/02x is fitted with one Flash EPROM part, namely an Intel 82802AC8 or equivalent Firmware Hub (FWH). This device is soldered to the board and is programmed at the factory with PC BIOS and factory test firmware. This EPROM will not normally be reprogrammed by the user, but Concurrent Technologies has programming software which allows BIOS updates to be carried out in the field when necessary, perhaps to add new features. Contact Concurrent Technologies for a copy of this software, and for the BIOS reprogramming information, if you believe that such an update is required.

Also programmed into the FWH at the factory is a Recovery BIOS, which allows the board to be restarted in a basic functional mode even if the main BIOS firmware has been corrupted. See Section 10.7 for further details of this feature.

The bottom half of the FWH device is used by Concurrent Technologies for storage of factory test firmware. The user may overwrite the factory test firmware if desired, but should be aware that it may be re-instated if the board is ever returned to Concurrent Technologies for repair or upgrading.

## 8.2 DRAM

The PP 33x/02x board supports a large amount of ECC or non-ECC DDR SDRAM. Two 200-pin SODIMM socket sites allow one or two modules of up to 1 Gbyte capacity each to be fitted either at the factory or in the field, giving a maximum size of 2 Gbytes. Section 2.6 describes how to fit the SODIMM, and details the types supported. The DRAM can be accessed from the processor, the local PCI busses and the CompactPCI backplane.

This page is intentionally unused.

# 9 ADDITIONAL LOCAL I/O FUNCTIONS

The PP 33x/02x supports a variety of I/O functions whose addresses are summarized in Table 9-1.

**NOTE:** An LPC Super I/O controller (designated TM Super I/O in Table 9-1) is located on the AD PP5/001-3x Transition Module. This device provides some of the legacy peripheral functions

**NOTE:** The Super I/O controller, and its associated legacy peripheral functions, are not available if the second serial port option is fitted.

| I/O Address Range | Description |
|---|---|
| 0000-001Fh | DMA Controller (6300ESB) |
| 0020-002Dh | Interrupt Controller (6300ESB) |
| 002E-002Fh | Config'n Index & Data Reg's (TM Super I/O) |
| 0030-003Dh | Interrupt Controller (6300ESB) |
| 0040-0043h | PIT Timers (6300ESB) |
| 004E-004Fh | SIU Config'n Index & Data Reg's (6300ESB) |
| 0040-0053h | PIT Timers (6300ESB) |
| 0060h | Keyboard Controller emulation (CPLD) |
| 0061h | NMI Status (6300ESB) |
| 0064h | Keyboard Controller emulation (CPLD) |
| 0070h | NMI Enable/RTC Address (6300ESB) |
| 0071h | RTC Data (6300ESB) |
| 0080h | Debug Port |
| 0092h | Port 92 Reset Generator (6300ESB) |
| 00A0-00A1h | Slave Interrupt Controller (6300ESB) |
| 00C0-00DFh | Slave DMA Controller (6300ESB) |
| 00F0h | Math Coprocessor Error |
| 0210-021Fh | Control & Status Registers (CPLDs) |
| 02F8-02FFh | COM2 Serial (TM Super I/O) |
| 03BC-03BFh | Parallel Port LPT1 (TM Super I/O) |
| 03E8-03Efh | COM3 Serial (TM Super I/O) |
| 03F0-03F7h | Floppy Controller (TM Super I/O) |
| 03F8-03FFh | COM1 Serial (6300ESB) |
| 04D0-04D1h | Interrupt Control (6300ESB) |
| 0CA9-0CABh | IPMI SMIC Interface (CPLD) |
| 0CF8-0CFFh | PCI Configuration Registers (855GME) |
| 0D00-FFFFh | PCI Free I/O Space |

**Table 9-1 I/O Address Map**

**NOTE:** I/O addresses in the range 0000h-0CFFh which are not listed above should not be accessed. The effects of such accesses are unpredictable.

# ADDITIONAL LOCAL IO FUNCTIONS

Most of the addresses are standard PC-AT compatible values, but at addresses 0210h - 021Fh and 0CA9h – 0CABh the board provides custom Status and Control registers for the board specific features.

There are 9 byte wide Status and Control registers. They fall into two groups, namely, general-purpose registers and IPMI SMIC interface registers. They are accessed at the following addresses:

- 210h for Status & Control Register 0
- 212h for Status & Control Register 1
- 213h for General Purpose I/O Register
- 214h for Status & Control Register 2
- 215h for Interrupt Control Register
- 21Dh for Interrupt Configuration Register
- 0CA9h for SMIC Data Register
- 0CAAh for SMIC Control/Status Register
- 0CABh for SMIC Flags Register

**NOTE:** All other I/O addresses in the range 0210h – 021Fh are reserved. Do not write to any of these addresses.

## 9.1 Status & Control Register 0

This register is at I/O address 210h.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RSVD | USER LED | ENUM# INTERRUPT ENABLE | FORCE SATELLITE | CPCI 66 MHz | PMC 66 MHz | PCI_PRESENT | CONS PORT |

**Bit 0: Console Port**
This bit indicates the setting of the Console Port Switch (see Section 6.1.2).

0 = COM1
1 = COM2

**Bit 1: CompactPCI PCI_PRESENT# Pin Status (Read Only)**
This bit indicates whether or not the PCI bus is present at this backplane slot.

0 = PCI bus not present
1 = PCI bus present

**Bit 2: PMC PCI Bus Frequency (Read Only)**
0 = 33 MHz
1 = 66 MHz

**Bit 3: CompactPCI Bus Frequency (Read Only)**
0 = 33 MHz
1 = 66 MHz

**Bit 4: Force Satellite Mode (Read Only)**
This bit indicates the state of the Satellite Mode switch (see Section 2.10). This switch is used to force Satellite Mode irrespective of which backplane slot the board is plugged into.

0 = Normal Mode selection
1 = Force Satellite Mode

**Bit 5: Enable Interrupt Generation from ENUM# (Read/Write)**
0 = disable Interrupt generation
1 = enable Interrupt generation

**Bit 6: User LED (Read/Write)**
0 = User LED Off
1 = User LED On

**Bit 7: Reserved**

## 9.2 Status & Control Register 1

This register is at I/O address 212h.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| FP NMI | IPMI NMI | FAL# | DEG# | MODE | CONSOLE | USER | SYSEN# |

**Bit 0: CompactPCI SYSEN# Pin Status (Read Only)**
This bit indicates whether or not the board is plugged into a System Controller slot.

0 = not System Controller
1 = System Controller

**Bit 1: User Switch (Read Only)**
Available for user defined purposes when the board starts up in BIOS mode (see Section 10.1). In CPSA (factory test) mode this switch selects between MTH and Soak operation.

0 = MTH
1 = Soak

**Bit 2: Console Switch (Read Only)**
Used to define the BIOS default standard input/output mode.

0 = input/output via serial port
1 = input via keyboard/output via VGA adapter

**Bit 3: Mode Switch (Read Only)**
Used to define the operating mode following a reset.

0 = BIOS operation
1 = CPSA operation

**Bit 4: CompactPCI 'DEG#' Signal is the cause of NMI (Read Only)**
0 = event has not occurred
1 = event has occurred

**Bit 5: CompactPCI 'FAL#' Signal is the cause of NMI (Read Only)**
0 = event has not occurred
1 = event has occurred

**NOTE:** DEG# & FAL# only generate an NMI when first asserted. They will not generate another interrupt until they have cycled false then true again. Bits 4 and 5 can be used as monitoring bits for these signals as they may be asserted for some time. The clearing of these bits is PSU dependant and beyond the scope of this document.

**Bit 6: IPMI is the cause of NMI (Read/Clear)**
0 = event has not occurred
1 = event has occurred

Writing zero to this bit will clear it to zero, writing one will leave it unchanged.

**Bit 7: Front Panel Switch is the cause of NMI (Read/Clear)**
0 = event has not occurred
1 = event has occurred

Writing zero to this bit will clear it to zero, writing one will leave it unchanged.

## 9.3 General Purpose I/O Register

This register is at I/O address 213h.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| EXT RESET | FAN SENSOR | INPUT 1 | INPUT 0 | RFU | USB POWER | OUTPUT 1 | OUTPUT 0 |

**Bits 1 - 0: General Purpose Outputs to Transition Module (Read/Write)**
0 = set output line to 0
1 = set output line to 1

**Bit 2: Transition Module USB Power Enable (Read Only)**
0 = TM USB power is Off
1 = TM USB power is On

**Bit 3: Reserved**

**Bits 5 - 4: General Purpose Inputs from Transition Module (Read Only)**
0 = input line is at 0
1 = input line is at 1

**Bit 6: Fan Sensor Status from Transition Module (Read Only)**
0 = input line is at 0
1 = input line is at 1

**Bit 7: Ext Reset Status from Transition Module (Read Only)**
0 = input line is at 0
1 = input line is at 1

**NOTE:** A multiplexed serial I/O scheme is used to connect these register bits to the I/O pins on the Transition Module. Because of this, there is an output latency of 64µs and the maximum input frequency (to avoid losing input transitions) is approximately 8 kHz.

## 9.4 Status & Control Register 2

This register is at I/O address 214h.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| PMC 1 | REV2 | REV1 | REV0 | CPVIO OK | RFU | EREADY | PMC 2 |

**Bit 0: PMC site 2 PMC Mode Status (Read Only)**
0 = PCI compliant module not fitted
1 = PCI compliant module fitted

**Bit 1: Processor PMC Enumeration Status (Read Only)**
0 = Processor PMC module(s) ready for enumeration
1 = Processor PMC module(s) not ready for enumeration

**Bit 2: Reserved**

**Bit 3: CompactPCI V(I/O) Voltage Status (Read Only)**
0 = voltage not OK
1 = voltage OK

This bit reports the status of the CompactPCI bus V(I/O) voltage. It reads 1 if the V(I/O) voltage at the backplane is greater than 3V.

**Bit 6-4: Hardware Revision Strapping (Read Only)**
000 = Rev A,
001 = Rev B etc…

**Bit 7: PMC site 1 PMC Mode Status (Read Only)**
0 = PCI compliant module not fitted
1 = PCI compliant module fitted

## 9.5 Interrupt Control Register

This register is at I/O address 215h. It provides control over interrupts from the IPMI.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RFU | GPI INT FLAG | SMS_ATN INT FLAG | SMIC NOT BUSY INT FLAG | FRU | GPI INT ENA | SMS_ATN INT ENA | SMIC NOT BUSY INT ENA |

**Bit 0: SMIC Not Busy Interrupt Enable (Read/Write)**
This bit allows an interrupt to be generated when the IPMI microcontroller clears the SMIC BUSY flag.

0 = interrupt disabled
1 = interrupt enabled

**Bit 1: SMS_ATN Interrupt Enable (Read/Write)**
This bit allows an interrupt to be generated when the IPMI microcontroller sets the SMIC SMS_ATN bit.

0 = interrupt disabled
1 = interrupt enabled

**Bit 2: General Purpose Interrupt (GPI) Interrupt Enable (Read/Write)**
This bit allows an interrupt to be generated when the IPMI microcontroller sets the GPI INT FLAG bit.

0 = GPI not enabled
1 = GPI enabled

**Bit 3: Reserved**

**Bit 4: SMIC Not Busy Interrupt Flag (Read/Clear)**
0 = event has not occurred
1 = event has occurred

Writing zero to this bit will clear it to zero, writing one will leave it unchanged.

**Bit 5: SMS_ATN Interrupt Flag (Read/Clear)**
0 = event has not occurred
1 = event has occurred

Writing zero to this bit will clear it to zero, writing one will leave it unchanged.

**Bit 6: General Purpose Interrupt (GPI) Interrupt Flag (Read/Clear)**
The IPMI microcontroller sets this bit to request an interrupt. If the GPI Interrupt Enable bit is also set, an interrupt (INT 5) will be generated. Note that if the interrupt is not required, the IPMI microcontroller can use this bit to signal status to the processor.

0 = no interrupt request
1 = interrupt request

Writing zero to this bit will clear it to zero, writing one will leave it unchanged.

**Bit 7: Reserved**

## 9.6 Interrupt Configuration Register

This register is at I/O address 21Dh.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| IPMI INTERRUPT STATUS | DEG# & FAL# MASK | PIT INTERRUPT STATUS | PIT INTERRUPT ENABLE | M66EN INTERRUPT FLAG | M66EN INTERRUPT ENABLE | ENUM# STATUS | ENUM# ROUTING |

**Bit 0: CompactPCI ENUM# Interrupt Routing (Read/Write)**
This bit selects which interrupt the CompactPCI ENUM# signal is routed to. This interrupt is only relevant when the board is System Controller.

0 = NMI
1 = PCI bus interrupt

**NOTE:** Bit 5 of Status and Control Register 0 enables this interrupt.

**Bit 1: CompactPCI ENUM# Pin Status (Read Only)**
0 = ENUM# is not asserted
1 = ENUM# is asserted

**Bit 2: CompactPCI M66EN High-Low Transition Interrupt Enable (Read/Write)**
When the board is System Controller it monitors the CompactPCI M66EN signal for high to low transitions during normal operation. A PCI bus interrupt may be generated if such a transition occurs. This bit enables that interrupt. The setting of bit 0 has no effect on this interrupt.

0 = transition interrupt is disabled
1 = transition interrupt is enabled

**Bit 3: CompactPCI M66EN High-Low Transition Flag (Read/Clear)**
This flag can be cleared by writing to the register with a zero in this bit position.

0 = clear flag
1 = leave flag unchanged

**Bit 4: PIT Interrupt Enable (Read/Write)**
This bit allows an interrupt to be generated when the PIT_INT signal from the 6300ESB ICH periodic interrupt timer changes state.

0 = PIT interrupt is enabled
1 = PIT interrupt is disabled

**Bit 5: PIT Interrupt Flag (Read/Clear)**
This flag can be cleared by writing to the register with a zero in this bit position.

0 = clear flag
1 = leave flag unchanged

**Bit 6: DEG# and FAL# Mask**
This bit allows the interrupt from the CompactPCI DEG# and FAL# signals to be masked (i.e. disabled).

0 = enable interrupt
1 = mask interrupt (power-on default)

**NOTE:** On earlier revisions of the board this bit was RFU.

**Bit 7: IPMI Interrupt Status (Read Only)**
0 = IPMI interrupt is not asserted
1 = IPMI interrupt is asserted

## 9.7 IPMI SMIC Interface

The IPMI hardware on this board is accessed by the local CPU using the SMIC interface (see Chapter 7 for an explanation of these terms and of the purpose of IPMI). The following sections outline the register contents, and example code for using this interface is provided in Section 7.6.

### 9.7.1 SMIC Data Register

This register is at I/O address 0CA9h. It is dual-ported. Both the CPU and the IPMI microcontroller can read it or write to it. The SMIC software protocol ensures that no contentions will occur.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |

**Bits 7-0: SMIC Data Value (Read/Write)**

### 9.7.2 SMIC Control/Status Register

This register is at I/O address 0CAAh. It is dual-ported. Both the CPU and the IPMI microcontroller can read it or write to it. The SMIC software protocol ensures that no contentions will occur.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| CS7 | CS6 | CS5 | CS4 | CS3 | CS2 | CS1 | CS0 |

**Bits 7-0: SMIC Control/Status Value (Read/Write)**

### 9.7.3 SMIC Flags Register

This register is at I/O address 0CABh. It reports the status of various SMIC flag bits.

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| RX_DATA_RDY | TX_DATA_RDY | RFU | SMI | EVT_ATN | SMS_ATN | RFU | BUSY |

**Bit 0: BUSY (Read/Set)**
**Bit 1: Reserved**
**Bit 2: SMS_ATN (Read Only)**
**Bit 3: EVT_ATN (Read Only)**
**Bit 4: SMI (Read Only)**
**Bit 5: Reserved**
**Bit 6: TX_DATA_RDY (Read Only)**
**Bit 7: RX_DATA_RDY (Read Only)**

# ADDITIONAL LOCAL IO FUNCTIONS

## 9.8 P.O.S.T. LED / Speaker

The P.O.S.T. LED is controlled via the speaker port. The P.O.S.T. LED replaces a PC speaker and is programmed in the same way a speaker would be programmed. The board also outputs the speaker port via a high current open collector driver on the CompactPCI J5 connector for connection to an external speaker if required.

## 9.9 PORT 80

A header (S2 – see Figure A-1) has been provided for monitoring data written to I/O Port 80. The PC BIOS writes status bytes to Port 80 that indicate a boot progress status and/or highlight any faults found. Data written to this port can be monitored using a Logic State Analyzer (LSA) or seven segment hexadecimal displays. See Section A.5.8 for details of the connector used for this port.

After boot-up this port can be used to monitor other status bytes written to port 80, which can be useful for debug purposes.

This page is intentionally unused.

# 10 PC BIOS

The PP 33x/02x board is fitted with PC BIOS firmware that performs many of the functions of a standard desktop PC. It also includes additional features specifically tailored for the CompactPCI bus environment. In addition to the core BIOS firmware, the board is fitted with BIOS Extensions for remote boot-load capability via any of the on-board Ethernet channels. To improve the flexibility of the board, some of these features may be selectively enabled or disabled by an operator using BIOS setup menus. Many of the features provided by the PC BIOS are unlikely to be adjusted by the user, but there are several options which many users will find helpful. Some of these are already referenced in other sections of this manual, but the remainder of this chapter will describe some other commonly used options. More information about each of the options available is provided in the Help box of the BIOS setup menus.

## 10.1 Entering the PC BIOS

The startup mode of the board may be selected using the Boot Mode switch, but can be either of the following:- PC BIOS mode (the factory default setting), which generally follows the behavior of a desktop PC, or CPSA mode (a flexible testing mode primarily for use at the factory), which can be used for system or board testing. CPSA mode operation and features are not described in this manual. Figure 10-1 shows the location of the Boot Mode switch on the board and its settings.

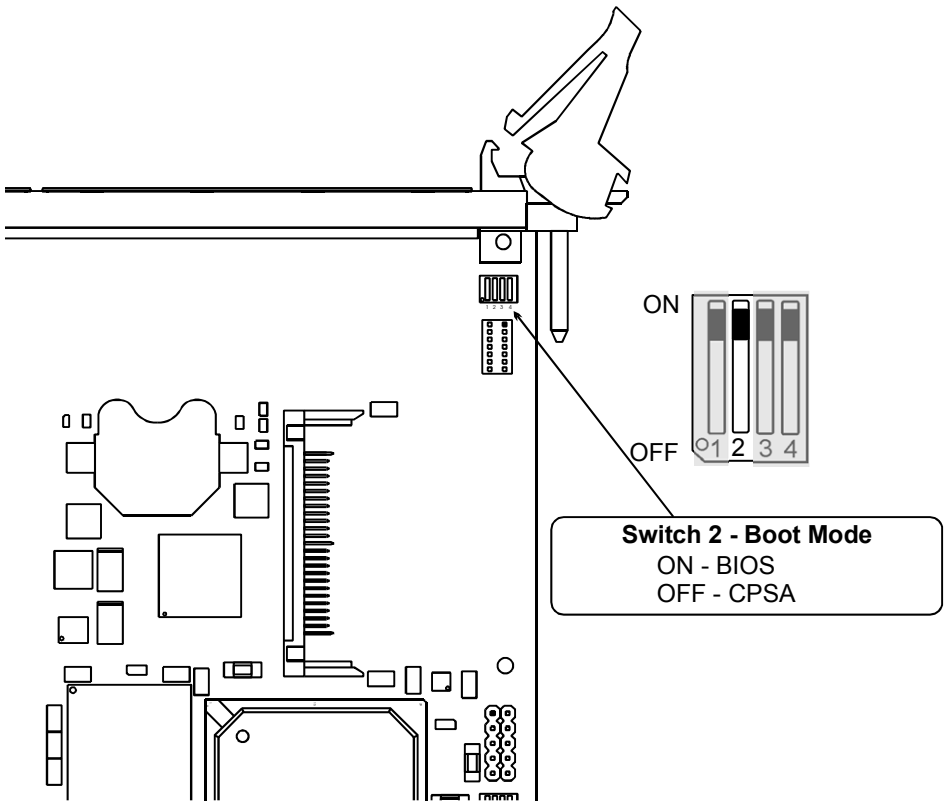



**Figure 10-1 Boot Mode Switch**

CPSA mode may be exited either by operator command, or by allowing the board to proceed through the CPSA startup sequence without intervention. In either case, the board will enter PC BIOS mode and continue as if this mode had been selected with the option switch. When the board is reset, it will generally restart in the operating mode selected by the Boot Mode option switch. However, a reset caused by a keyboard <CTRL-ALT-DEL> keystroke combination, or by a programmed reset using one of several different I/O access sequences, will only cause a PC BIOS restart. A complete board or system reset (using the front panel switch or through the CompactPCI bus PCI_RST signal) will cause the board to restart in the mode selected by the Boot Mode option switch setting.

Operator communication with the PC BIOS is usually through a serial terminal connected to the COM1 or COM2 serial ports. This can be reconfigured with a board option switch to use a VGA display (either using the on-board graphics interface, if fitted, or via a PMC module) and a separate keyboard. Section 6.1.1 describes the location and settings for this switch. Selection between COM1 and COM2 is done via a board option switch. Section 6.1.2 describes the location and settings for this switch.

A VT100-compatible serial terminal or emulator program should be used. By default the serial line is programmed to operate at 9600 Baud with 8 data bits, 1 stop bit and no parity (8N1). There is no flow control. For fast terminals, the baud rate can be increased via the **`Serial Console Baud Rate`** field of the **`Main`** Setup menu.

## 10.2 The PC BIOS Startup Sequence

When the board starts up without operator intervention, it will run a basic Power-On Self-Test (POST) sequence, including ECC DRAM initialization and a DRAM test. The full DRAM test will be omitted on subsequent restarts if the BIOS configuration settings have not been changed. Once the DRAM test has completed, the board will try to bootload application software from any attached mass storage medium or through one or both of the Ethernet interfaces.

**WARNING:** If the V(I/O) supply voltage is not wired on the backplane, the CompactPCI bridge drive will not operate correctly. As a result, the CompactPCI bus device scan will cause the board to hang.

When the PC BIOS starts after changing the battery, losing battery power or after using the CMOS CLEAR jumper, it may report a CMOS Checksum Error or some other problem. This will be following by a prompt to the operator to press <F1> to continue or <F2> to enter Setup mode. If no key is pressed within approximately five seconds, the PC BIOS will continue with its normal startup sequence. It will also re-calculate the CMOS Checksum to prevent this error occurring again at a subsequent restart.

Pressing the <F2> key at any time during the PC BIOS startup sequence will result in the BIOS Setup menu being entered. The Setup menu is quite extensive, and is provided with context-sensitive help information, which is displayed in the right-hand panel on screen.

**NOTE:** When the <F2> key is pressed, a few seconds may elapse before the BIOS Setup menu appears. The PC BIOS will always run BIOS Extensions for any PMC modules it detects before responding to the key press.

## 10.3 Boot Device Selection

The order in which the PC BIOS searches for a bootable medium is pre-configured but may be altered by the operator using the **Boot** setup menu. When the order is changed using this menu it will be retained in non-volatile memory so that the order is maintained after a restart. It is also possible to specify a one-time override of the boot device when the board starts, by pressing the <ESC> key. This will result in a pop-up menu appearing. The appropriate boot device may be selected from a list by using the cursor keys and pressing <ENTER>, but this is not retained in non-volatile memory, so the correct device must be re-selected if necessary at a subsequent restart.

**NOTE:** When the <ESC> key is pressed, a few seconds may elapse before the boot device selection menu appears. The PC BIOS will always run BIOS Extensions for any PMC modules it detects before responding to the key press.

The on-board Ethernet channels require their PXE Firmware to be enabled before they can be used as boot devices. A BIOS setup option controls whether PXE Firmware runs for the front or rear Ethernet channels, this can be found under **Main | Boot Options | PXE Boot Firmware**.

The Ethernet boot firmware allows remote booting using the Pre-Boot Execution (PXE) protocols. Further information on the capabilities of this software is available from the Intel web site at:

http://developer.intel.com

**NOTE:** The BIOS has limited space available for Extension ROMs. If a PMC module containing extension firmware is fitted to the board, it may be necessary to disable one or more of the on-board firmware extensions before the PMC firmware can be loaded.

The BIOS runs Extension ROM firmware for each PCI device matching a ROM's signature. This means that if the BIOS detects two or more devices of the same type it will attempt to load and run the same firmware more than once. Since the space available for Extension ROMs is limited, this process can result in error messages being issued during POST (and the side effect of running the full memory test each time the board boots).

To overcome this problem the BIOS includes a Setup option**, Main | Boot Options | Option ROM Loading**, which controls the number of times each ROM Extension is run. By default there is no restriction, however the user can elect to run SCSI Extension ROMs once only, or run all Extension ROMs once only.

## 10.4 PCI Bus Resource Management

The bus structure of the PP 33x/02x is complex. There are two on-board PCI busses: a 64-bit bus which connects the 6300ESB ICH to the 82546GB Ethernet controller, the PMC sites and the CompactPCI bridge chip; and a 32-bit bus which connects the 6300ESB ICH to the two 82541PI Ethernet controllers.

The 32-bit bus operates at 33 MHz with 3.3V signaling levels. The 64-bit bus operates with 5V or 3.3V signaling levels and normally runs at 66 MHz. It may be slowed to 33 MHz if a 33 MHz PMC module is fitted, or by setting a board switch (see Section 2.9.1).

### 10.4.1 PCI Resource Allocation

The PC BIOS initializes all devices on the local PCI bus, and allocates appropriate memory address ranges, I/O address ranges, and interrupt routings for all these devices. This process is automatic as part of the BIOS "Plug-and-play" setup. Devices on the CompactPCI bus may also have memory, I/O or interrupt resources, these will also be configured by the PC BIOS. The Intel chipset allows for a flexible allocation of many PCI bus interrupts to the available interrupt inputs on the PC-compatible interrupt controllers provided on the board. The PC BIOS uses this feature to program default settings which it considers appropriate for the combination of on-board devices and any device fitted to the PMC site. In some configurations, depending on the operating system being used and the capability of the relevant device drivers, it may be necessary for the user to modify this default configuration, to minimize the sharing of interrupt lines. The PC BIOS Setup screen for **`Advanced | PCI Device Configuration`** allows this.

This screen allows the user to override the PC BIOS default selections for interrupt allocation, but care must be taken when doing this to avoid conflicts which may result in operating system or even BIOS "crashes". To allow maximum flexibility of choice for the user, the PC BIOS performs limited checks on the user's interrupt allocation. In the event that there is a problem, it may be necessary to clear the CMOS memory (see Section 2.8), or even to reset the Extended System Configuration Data via the **`Reset Configuration Data`** field of the BIOS Setup screen for **`Advanced`** configuration settings. The PC BIOS does not allow the user to override the allocation of memory and I/O address ranges.

**NOTE:** When reallocating interrupts using the BIOS Setup screens, try to avoid allocating the PMC interrupts to ones also allocated to other devices. This sharing of interrupts can cause problems with some operating systems where device drivers do not correctly handle shared interrupts.

Table 10-1 lists the configurable interrupts for this board. The actual allocation of PCI bus interrupts to available interrupt controller inputs will depend on both the default “Plug-and-play” settings programmed by the PC BIOS, and the way in which the user has overridden them using the Setup screens. When more than one PCI bus interrupt is routed to the same interrupt controller input, that input will remain active while any of the sources connected to it are active.

| |
|---|
| CompactPCI INTA |
| CompactPCI INTB |
| CompactPCI INTC |
| CompactPCI INTD |
| PMC Site 1 INTA |
| PMC Site 1 INTB |
| PMC Site 1 INTC |
| PMC Site 1 INTD |
| PMC Site 2 INTA |
| PMC Site 2 INTB |
| PMC Site 2 INTC |
| PMC Site 2 INTD |
| Ethernet Channel 0 |
| Ethernet Channel 1 |
| Ethernet Channel 2 |
| Ethernet Channel 3 |
| CompactPCI M66EN transition |
| CompactPCI ENUM# |
| 6300ESB PIT |
| IPMI Microcontroller |

**Table 10-1 Configurable PCI Bus Interrupts**

**NOTE:** CompactPCI Interrupts INTA - INTD are only relevant when the board is System Controller.

### 10.4.2 PCI Device IDs

Each PCI bus, and each device on an individual PCI bus, has a unique ID. For the PP 33x/02x, the bus and device IDs are listed in Table 10-2. The 855GME GMCH to 6300ESB ICH Hub Interface appears as PCI bus 0 to the processor. The PC BIOS reserves bus #1 for the 855GME GMCH virtual AGP bridge. The PCI busses are accessed via PCI to PCI bridges in the 6300ESB ICH device and will appear as busses 2 and 3 if no PMC modules are fitted. If PMC modules with on-board bridges are fitted, then bus 3 will move to a higher bus number.

| PCI Bus Number | PCI Device ID | PCI Function Code | Device Name Description |
|---|---|---|---|
| 0 | 0 | | 855GME GMCH |
| 0 | 2 | 0 | 855GME Graphics Controller |
| | | 1 | 855GME Graphics Controller |
| 0 | 28 - 31 | | 6300ESB ICH |
| | 28 | 0 | Hub Interface to PCI-X Bridge |
| | 29 | 0 | USB Controller #1 |
| | | 1 | USB Controller #2 |
| | | 4 | Watchdog Timer |
| | | 5 | IOxAPIC (not used) |
| | | 7 | USB 2.0 Controller |
| | 30 | 0 | Hub Interface to PCI Bridge |
| | 31 | 0 | PCI to LPC Bridge |
| | | 1 | EIDE Controller |
| | | 2 | SATA Controller (not used) |
| | | 3 | SMBus Controller |
| | | 5 | AC'97 Audio Controller (not used) |
| | | 6 | AC'97 Modem Controller (not used) |
| 1 | | | 855GME GMCH Virtual AGP Bridge |
| 2 | 1 | 0 | Ethernet Channel 2 (82546GB ch A) |
| | | 1 | Ethernet Channel 3 (82546GB ch B) |
| 2 | 7 | 0 | PMC Site 1 |
| 2 | 8 | 0 | PMC Site 2 |
| 2 | 12 | 0 | CompactPCI Bridge |
| 3 | 1 | 0 | Ethernet Channel 0 (82541PI 0) |
| 3 | 2 | 0 | Ethernet Channel 1 (82541PI 1) |

**Table 10-2 PCI Device Numbers**

## 10.5 CompactPCI Bridge Configuration

The BIOS provides Setup menus that allow configuration of certain features of the CompactPCI Bridge, these can be found under the **CompactPCI** top level menu. The fields available via Setup depend on the board's operating mode.

### 10.5.1 System Controller Mode

In System controller mode two Setup options are provided:

**ENUM Interrupt** allows the user to specify whether the CompactPCI ENUM interrupt drives NMI or IRQ5.

**M66EN Interrupt** allows the user to specify whether the CompactPCI M66EN signal drives IRQ5.

### 10.5.2 Satellite Mode

In Satellite mode the board does not connect to the CompactPCI interface, therefore no Setup menu is provided.

### 10.5.3 Peripheral Mode

**NOTE:** Peripheral mode is only available on Rev D boards and on selected variants of Rev C and Rev B boards, i.e. those fitted with the PCI 6254 CompactPCI bridge.

When the board is operating in Peripheral mode, the BIOS provides two sub-menus for configuring the upstream and downstream windows of the CompactPCI bridge.

#### 10.5.3.1 Upstream Windows

Upstream windows give the Peripheral mode PP 33x/02x board access to resources on the CompactPCI bus. The CompactPCI bridge provides three upstream windows: upstream window 0 can map either memory or I/O addresses; windows 1 and 2 are for memory addresses only.

The BIOS Setup menu allows the user to specify the size of each window and the CompactPCI address that will be accessed via the window. The local address of the window is configured automatically by the BIOS.

**NOTE:** The local address of upstream windows is not fixed and may change if the board configuration is modified. Application software should always read a window's base address from PCI configuration space.

#### 10.5.3.2 Downstream Windows

Downstream windows give devices on the CompactPCI bus access to resources on the Peripheral mode PP 33x/02x board. The CompactPCI bridge provides three downstream windows: downstream window 0 can map either memory or I/O addresses; windows 1 and 2 are for memory addresses only.

The BIOS allows the user to specify the size of each window and the local address that will be accessed via the window. The System Controller assigns the window's address on the CompactPCI bus.

### 10.5.4 Peripheral Mode Window-Size Limitations

The CompactPCI bridge forms PCI addresses by concatenating the least significant bits from the CPU generated address and the most significant bits from the translation basse address; the contribution from each part is fixed and depends upon the window type. However, the BIOS always aligns base addresses according to their resource size, to achieve optimal packing.If a window is defined to be smaller than the translation base address granularity, the BIOS assigned base address may result in configuration where offset 0h into the resource window does not map to offset 0h from the translation base address. The example below illustrates this point:

| | | |
|---|---|---|
| Configured I/O window size: | 256 bytes | |
| Bridge window granularity: | 4096 bytes | |
| BIOS assigned base address: | 12345600h | (i.e. 256 byte aligned) |
| Translation base address: | ABCDE000h | (i.e. 4096 byte aligned) |
| Address generated by bridge: | ABCDE6xxh | (xx = offset into window) |

In this example, accesses to offset 0h into the window result in accesses to offset 600h from the translated base address.

#### 10.5.4.1 I/O Windows

For I/O windows, the translation base address granularity, and hence minimum practical size, is 4Kbytes.

#### 10.5.4.2 Memory Mapped Windows

For memory mapped windows, the translation base address granularity, and hence minimum practical size, is 1Mbyte.

## 10.6 User Selectable NVRAM Defaults

The BIOS provides a facility through which the user can save preferred setup option settings to Flash memory (NVRAM). Then, if the BIOS detects that the contents of NVRAM is corrupt, the user can elect to restore the contents from the saved settings, rather than loading factory configured defaults. This facility also allows the board to operate without fitting the battery, but with NVRAM settings different to the factory defaults.

The **`Save User Defaults`** option in the BIOS Setup **`Exit`** menu is used to write the current NVRAM settings to Flash memory. When settings are saved to Flash memory, the current BIOS date and time will also be stored; this allows the board to start operating with an appropriate date and time.

**NOTE:** Saving settings to Flash memory may take several seconds to complete.

The NVRAM restore feature is controlled by the BIOS Defaults option switch (see Figure 10-3). When the BIOS Defaults switch is in the “User” position and the BIOS detects that the NVRAM contents are corrupt, NVRAM settings will be restored from Flash memory (provided that valid settings have been saved). When the switch is in the “Factory” position, the factory-configured defaults will be used.

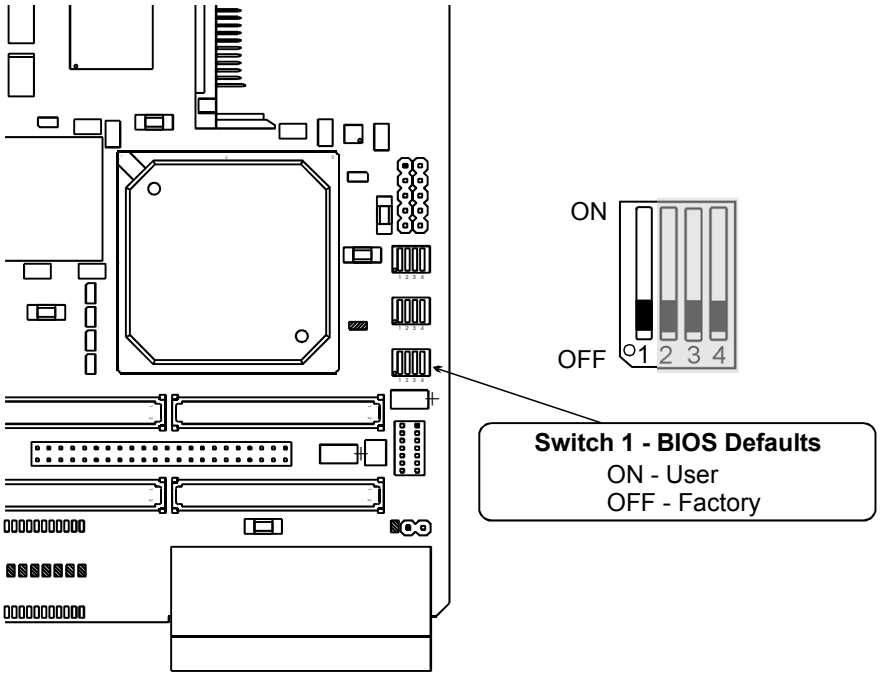



**Figure 10-2 BIOS Defaults Switch**

To configure and save preferred NVRAM settings:

- Configure the BIOS in the usual manner via the Setup menus
- Reboot the board and allow it to proceed through to the boot loader without error
- Reboot the board and enter Setup
- From the **`Exit`** menu, select **`Save User Defaults`**
- Set the BIOS Defaults switch to enable the restore feature

**NOTE:** The BIOS Defaults switch has a different function if the board is configured for CPSA (Factory Test) mode. If the board is ever returned to the factory for repair, this switch will be changed and the User Selectable NVRAM settings may be affected.

## 10.7 The Recovery BIOS

In the unlikely event that the board's BIOS ROM contents becomes corrupted and it is not possible to perform the normal BIOS update procedure, the board provides a minimal Recovery BIOS that will allow the board to boot from a specially prepared floppy disk and restore a known-good BIOS image.

The Recovery BIOS is located in a special sector in the BIOS ROM that is protected from accidental erasure by hardware means. When a BIOS update is performed, the Recovery BIOS does not get updated.

When power is applied to the board, or when the board is reset, the CPU starts to execute the Recovery BIOS. From here a checksum test is performed on the first 64Kbytes of the main BIOS to ensure that it is intact. When the checksum is validated, control passes to the main BIOS and the board boots normally. If the checksum test fails, the BIOS recovery process is invoked automatically.

The BIOS recovery process can also be forced using the Boot Type Switch, see Figure 10-3.

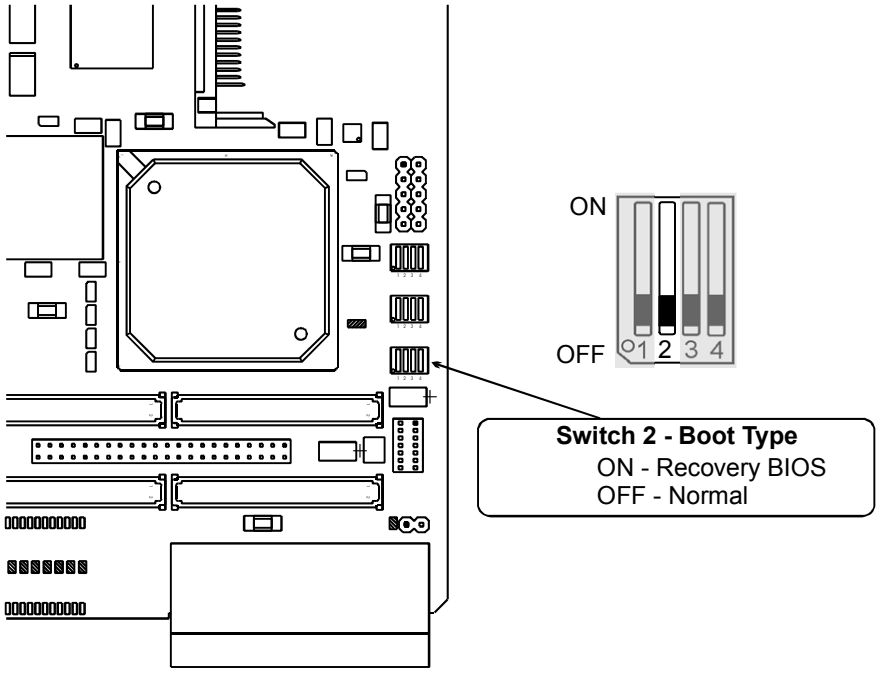



**Figure 10-3 Boot Type Switch**

The recovery BIOS will indicate its progress via a serial terminal connected to COM1 or COM2, as determined by the Console Part Switch (see figure 6-2). VGA output is not supported.

The recovery BIOS requires a specially prepared disk image: a suitable recovery disk image can be obtained from Concurrent Technologies, if required.

# 11 SYSTEM MANAGEMENT

## 11.1 Power Management

The Intel Pentium M processor incorporates a mechanism for changing the processor's operating frequency and core voltage under software control. By making these reductions the board's maximum power consumption is also reduced by as much as 45%.

The Intel Pentium M processor supports a number of discrete operating frequencies (typically five or six) that vary between 600 MHz and full speed (see Table 11-1). The BIOS provides a Setup menu (**`Advanced | CPU Settings | CPU Operating Frequency`**) that allows the operating frequency of the CPU to be set prior to booting an operating system. By default the CPU will operate at full speed.

| 1.8 GHz Intel Pentium M Processor 745 | 1.6 GHz Intel Pentium M Processor | 1.4 GHz Intel Pentium M Processor 738 |
|---|---|---|
| 1.8 GHz | 1.6 GHz | 1.4 GHz |
| 1.6 GHz | 1.4 GHz | 1.2 GHz |
| 1.4 GHz | 1.2 GHz | 1.0 GHz |
| 1.2 GHz | 1.0 GHz | 800 MHz |
| 1.0 GHz | 800 MHz | 600 MHz |
| 800 MHz | 600 MHz | |
| 600 MHz | | |

**Table 11-1 CPU Operating Speed BIOS Options**

This mechanism for power reduction allows the board to operate in environments where power capacity is limited (e.g. under battery power), or in systems where cooling airflow is less than adequate. In this last case it may be wise to also consider some of the Thermal Management options available on this board (see Section 11.2).

**NOTE:** During POST, the BIOS will always report the maximum possible CPU frequency, as specified by the CPU's Brand String.

## 11.2 Thermal Management

The maximum power dissipation of the Intel Pentium M processor may sometimes be higher than that of previous Mobile Intel Pentium processors. Under typical load conditions, the heatsink (and cooling airflow) will keep the processor die temperature within specification. However, if the board is running CPU-intensive or stress software or if the airflow is inadequate, the heatsink alone may not be able to prevent the processor overheating.

To ensure that the processor always operates within its thermal specifications, it includes several thermal management and protection functions. Each of these is described below. A BIOS setup option is used to select which functions are to be enabled. See Section 11.2.4 for further details.

### 11.2.1 Thermal Monitor 1 (TM1)

TM1 uses a temperature sensor located near to the hottest part of the processor die. If it detects a critically high temperature a thermal control circuit (TCC) will modulate (i.e. alternately stop and start) the processor core clocks. This causes the processor to halt for short periods and decreases its power consumption, which in turn lowers the die temperature.

The severity of the modulation will increase as the die temperature rises, up to a maximum of about 50%. The TCC will cease modulation when the die temperature has fallen to a non-critical value.

Intel individually calibrates the temperature sensor. The TM1 characteristics are also fixed by Intel and cannot be modified.

The drawbacks of TM1 are that it starts to operate at a relatively low temperature (about 90°C) and that it has a low modulation rate, which can produce undesirable software latencies.

TM1 is disabled after Reset and has to be enabled by the BIOS (see Section 11.2.4).

### 11.2.2 Thermal Monitor 2 (TM2)

TM2 uses the same temperature sensor as TM1. When the TM2 thermal control circuit is triggered, the CPU operating frequency and core voltage will be reduced, causing the power consumption to fall, which in turn lowers the die temperature.

Because a CPU under TM2 control operates continuously, the overall system performance for a given reduction in power consumption is higher than TM1. System latency is also lower when using TM2.

The drawback of TM2 is that changes to the operating frequency will also affect the Time Stamp Counter (TSC) rate and any timing or calculations that use it.

TM2 is disabled after Reset and has to be enabled by the BIOS (see Section 11.2.4).

### 11.2.3 CPU Thermal Trip

The processor chip also contains a thermal trip circuit. This is intended to protect the processor in the event of a catastrophic cooling failure. If the die temperature reaches approximately 135°C, it shuts down the processor core and asserts the THERMTRIP# signal. Logic on the board responds to this assertion by removing the processor core voltage within a few milliseconds.

A power cycle (i.e. OFF then ON) is required to restore normal operation. The thermal trip circuit is always operational and cannot be disabled.

**NOTE:** The User LED will flash rapidly (at a rate of approximately 4Hz) if the thermal trip circuit activates.

### 11.2.4 Ethernet Controller Thermal Considerations

On extended temperature boards there are several restrictions when operating the Ethernet interface at Gigabit speeds. These restrictions are discussed in Section 1.5

### 11.2.5 PC BIOS Setup Options

The PC BIOS Setup Menu provides control over the thermal management functions, using the setting for the **`Advanced | CPU Settings | Thermal Management`** option. It offers three choices, namely:

- Disabled - only the Thermal Trip is enabled
- Thermal Monitor 1 - the CPU's clock modulation mechanism is used to reduce power consumption; the Thermal Trip is enabled
- Thermal Monitor 2 - the CPU operating frequency is lowered to reduce power consumption; the Thermal Trip is enabled

The default setting is disabled.

It may be desirable that some form of thermal management is enabled to handle operation at very high load and high temperature. Under normal conditions, even with high loads, the standard heatsink and specified forced air cooling are adequate to keep the processor operating within its limits even with the Thermal Management option set to Disabled. The BIOS setup option provides a means to enable additional protection if there is any concern about heat dissipation in the particular system being used.

### 11.2.6 Processor Thermal Status Indication

The User LED may be programmed to indicate if the processor die has reached the critical temperature at which thermal management is activated. This is done using the User LED Mode option switch. Figure 11-1 shows the location of this switch on the board and its settings.

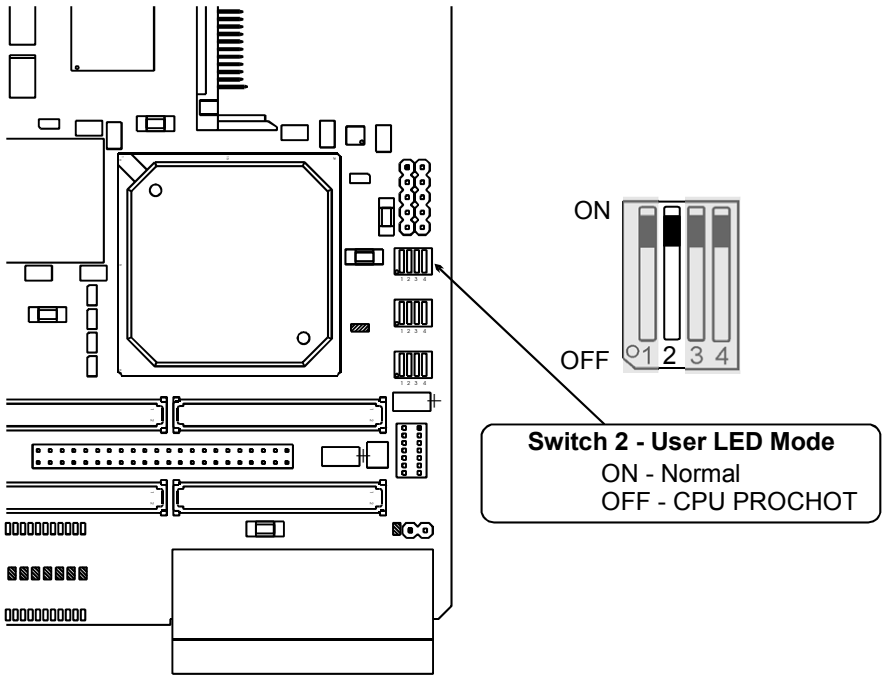



**Figure 11-1 User LED Mode Switch**

The User LED will also indicate a CPU thermal trip. See Section 11.2.3 for details.

## 11.3 ECC Error Logging

The 855GME North Bridge includes hardware for detecting and correcting single-bit ECC errors and for detecting multi-bit ECC errors. Both types of ECC error can be recorded by the BIOS via the IPMI System Event Log (SEL).

### 11.3.1 Single-bit ECC Errors

Single-bit ECC errors are a rare occurrence in normal operation. The chipset will detect single bit errors during memory reads and automatically pass the corrected data to the CPU instead.

Single-bit ECC errors are recorded by the BIOS so that the user can be made aware of any unusual behavior.

### 11.3.2 Multi-bit ECC Errors

Multi-bit ECC errors are a very rare occurrence in normal operation. The chipset can detect multi-bit errors, but is unable to make corrections; therefore the BIOS will halt the board to prevent invalid code or data being processed.

In the unlikely event that a board halts unexpectedly, the user should examine the event log to determine whether a multi-bit ECC error was the cause.

### 11.3.3 Background ECC Event Logging

The BIOS code responsible for recording ECC errors remains resident in System Management memory after the operating system loads. This code is invisible to the operating system and consumes no CPU cycles or memory resources, unless triggered.

The ECC error logging code is triggered by a System Management Interrupt, which is generated by the hardware ECC logic. The error logging code minimizes its impact on the operating system by only recording one single-bit ECC event per hour.

Multi-bit ECC errors will always cause the board to halt, regardless of whether ECC Event Logging is enabled.

To minimize processing time stolen from the operating system, the BIOS writes ECC error information to the local IPMI controller as a short OEM message. The local IPMI controller is then responsible for reformatting the data and adding it to the SEL. Single bit errors are identified by OEM SEL Record type C0h, multi-bit by C1h.

# A SPECIFICATIONS

## A.1 Functional Specification

| | |
|---|---|
| **Processor:** | • 1.8 GHz, 1.6 GHz or 1.4 GHz Intel Pentium M processor. |
| **Level 2 Cache:** | • 1024 Kbytes on-die RAM operating at core frequency (1.6 GHz processor).<br>• 2048 Kbytes on-die RAM operating at core frequency (1.4 GHz and 1.8 GHz processor). |
| **Memory:** | • 1 Mbyte Flash EPROM for PC BIOS using soldered 82802AC8 or equivalent Firmware Hub device.<br>• Up to 2 Gbytes DDR SDRAM as defined by order number. |
| **Interfaces:** | • 64/32-bit 66/33 MHz CompactPCI interface utilizing a PCI to PCI bridge. The board supports 5V or 3.3V CPCI signaling levels.<br>• One RS232 serial channel via in-board header using 16550 compatible UART.<br>• EIDE/Ultra ATA100 interface via J5, on-board Compact Flash site and on-board mass storage option interface.<br>• Two USB interfaces via front panel connector and one USB interface via J5 connector. Both USB 1.1 (1.5 and 12Mbit/s) and USB 2.0 (480Mbits/s) operation are supported.<br>• Two single-width PMC sites supporting 64/32-bit 66/33 MHz PCI interfaces with 3.3V signaling. Both 5V and 3.3V power rails are provided.<br>• Two Gigabit Ethernet interfaces using 82546GB Ethernet controller with 10/100/1000Mbits/s connections via J3. Support for PICMG 2.16 backplane networking. Optional rear panel Ethernet I/O via RJ45 connections on transition module.<br>• Two front panel Gigabit Ethernet interfaces using two 82541PI Ethernet controllers with 10/100/1000Mbits/s operation.<br>• External Push Button Reset input via J2.<br>• IPMI SMIC interface and IPMB 0 and IPMB 1 interfaces supported by on-board microcontroller. |
| **Peripherals:** | • Intel 6300ESB ICH device provides standard PC-AT architecture peripherals.<br>• PC AT Real Time Clock.<br>• 32-bit Periodic Interval Timer with processor interrupt capability. |

# SPECIFICATIONS

## A.2 Environmental Specification (N Series)

### A.2.1 Temperature Range

Operating..... 0ºC to +55ºC @ 400LFM air flow
Storage..... -40ºC to +85ºC

The processor die temperature can be monitored via the IPMI subsystem (see Section 7.4.4).

**NOTE:** If the on-board disk drive option is fitted, the operating temperature range will be restricted to +5ºC to +55ºC and the storage temperature range will be restricted to -40ºC to +65ºC.

**NOTE:** The battery life will be reduced by storage at high temperatures due to increased self-discharge. It is therefore recommended that the battery be removed during storage.

### A.2.2 Humidity

Operating..... 10% to 90% non-condensing
Storage..... 10% to 90% non-condensing

## A.3 Dimensions

| | |
|---|---|
| Height | 23.3cm |
| Depth | 16.0cm |
| Width | 2.0cm |
| Weight | 540g (without Mass Storage Kit fitted) |

# SPECIFICATIONS

## A.4 Electrical Specification

### A.4.1 Power Supply Requirements

| Full Speed | Actual Speed | +5V +5%, -3% | | +3.3V +5%, -3% | | +12V +/-5% | -12V +/-5% |
|---|---|---|---|---|---|---|---|
| | | Typical | Maximum | Typical | Maximum | Maximum | Maximum |
| 1.8 GHz | 1.8 GHz | 3.5A | 6.3A | 2.0A | 2.3A | 0.05A | 0.05A |
| | 1.2 GHz | 2.1A | 3.4A | 2.0A | 2.3A | 0.05A | 0.05A |
| | 600 MHz | 1.4A | 1.9A | 2.0A | 2.3A | 0.05A | 0.05A |
| 1.6 GHz | 1.6 GHz | 3.6A | 7.0A | 2.1A | 2.4A | 0.05A | 0.05A |
| | 1.2 GHz | 2.2A | 4.0A | 2.1A | 2.4A | 0.05A | 0.05A |
| | 600 MHz | 1.2A | 1.8A | 2.1A | 2.4A | 0.05A | 0.05A |
| 1.4 GHz | 1.4 GHz | 2.3A | 3.7A | 2.0A | 2.3A | 0.05A | 0.05A |
| | 1.0 GHz | 2.2A | 2.7A | 2.0A | 2.3A | 0.05A | 0.05A |
| | 600 MHz | 1.5A | 2.0A | 2.0A | 2.3A | 0.05A | 0.05A |

**Table A-1 Voltage and Current Requirements**

**NOTE:** The maximum current values for the +5V and 3.3V rails will not normally be reached simultaneously. In general, the +5V rail current will increase with CPU activity, and the 3.3V current with DRAM activity however caused, e.g. DMA transfers.

**NOTE:** This is for a board with 1 Gbyte SDRAM and with no Mass Storage Kit, PMC modules or CompactFlash fitted.

**NOTE:** +/- 12V supplies are provided primarily for the PMC interface but the +12V supply must be present for the board to operate correctly. When used for PMC they are limited to 500mA for +12V and 200mA for -12V.

## A.5 Connectors

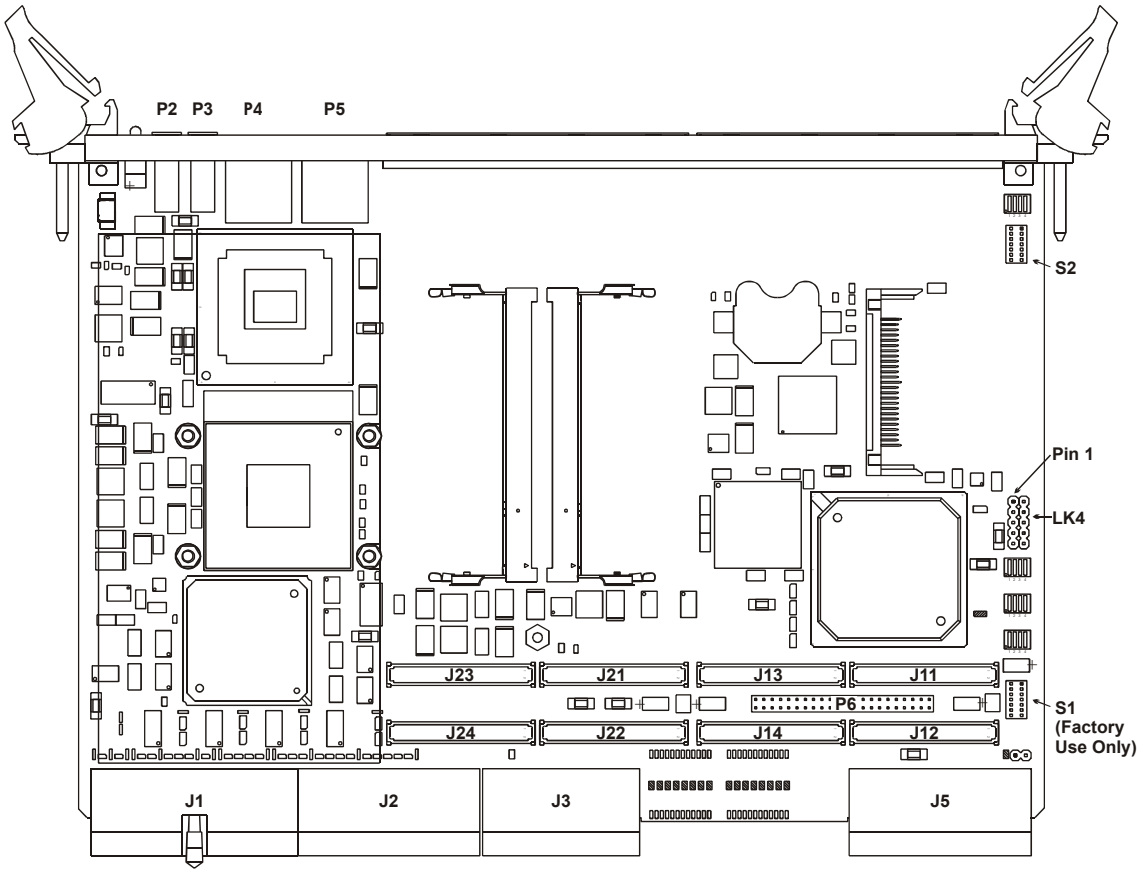



**Figure A-1 Connector Layout**

**NOTE:** On Rev A boards, the location of headers S1 and S2 are swapped over.

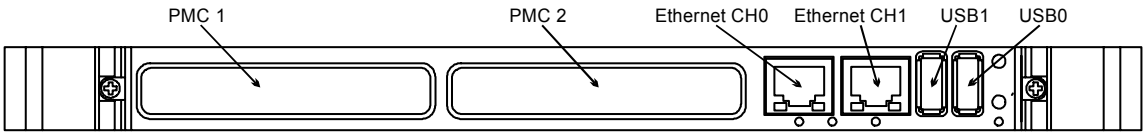



**Figure A-2 Front Panel Connectors**

### A.5.1 CompactPCI Interface (J1) Pin-outs

The CompactPCI interface connector J1 consists of a 150-pin connector with pins assigned as follows:

| Pin | A | B | C | D | E | F | |
|---|---|---|---|---|---|---|---|
| 25 | 5V | REQ64# | ENUM# | 3.3V | 5V | GND | |
| 24 | AD[1] | 5V | V(I/O) | AD[0] | ACK64# | GND | |
| 23 | 3.3V | AD[4] | AD[3] | 5V | AD[2] | GND | |
| 22 | AD[7] | GND | 3.3V | AD[6] | AD[5] | GND | |
| 21 | 3.3V | AD[9] | AD[8] | M66EN | C/BE[0]# | GND | |
| 20 | AD[12] | GND | V(I/O) | AD[11] | AD[10] | GND | |
| 19 | 3.3V | AD[15] | AD[14] | GND | AD[13] | GND | J |
| 18 | SERR# | GND | 3.3V | PAR | C/BE[1]# | GND | 1 |
| 17 | 3.3V | IPMB_SCL | IPMB_SDA | GND | PERR# | GND | |
| 16 | DEVSEL# | GND | V(I/O) | STOP# | LOCK# | GND | C |
| 15 | 3.3V | FRAME# | IRDY# | BD_SEL# | TRDY# | GND | O |
| 12-14 | KEY AREA | | | | | | N |
| 11 | AD[18] | AD[17] | AD[16] | GND | C/BE[2]# | GND | N |
| 10 | AD[21] | GND | 3.3V | AD[20] | AD[19] | GND | E |
| 9 | C/BE[3]# | IDSEL | AD[23] | GND | AD[22] | GND | C |
| 8 | AD[26] | GND | V(I/O) | AD[25] | AD[24] | GND | T |
| 7 | AD[30] | AD[29] | AD[28] | GND | AD[27] | GND | O |
| 6 | REQ0# | PCI_PRESENT# | 3.3V | CLK | AD[31] | GND | R |
| 5 | NC | NC | PCI_RST# | GND | GNT0# | GND | |
| 4 | IPMB_PWR | HEALTHY# | V(I/O) | INTP | INTS | GND | |
| 3 | INTA# | INTB# | INTC# | 5V | INTD# | GND | |
| 2 | NC | 5V | NC | NC | NC | GND | |
| 1 | 5V | -12V | NC | +12V | 5V | GND | |
| Pin | A | B | C | D | E | F | |

**Table A-2 CompactPCI J1 Interface Pin-outs**

**NOTE:** PICMG 2.16 defines pin B6 as PCI_PRESENT#. In earlier PICMG specifications this pin is defined as GND.

### A.5.2 CompactPCI Interface (J2) Pin-outs

The CompactPCI interface connector J2 consists of a 132-pin connector with pins assigned as follows:

| Pin | A | B | C | D | E | F | |
|---|---|---|---|---|---|---|---|
| 22 | GA4 | GA3 | GA2 | GA1 | GA0 | GND | |
| 21 | CLK6 | GND | NC | NC | NC | GND | |
| 20 | CLK5 | GND | NC | GND | NC | GND | |
| 19 | GND | GND | IPMB1 SDA | IPMB1 SCL | IPMB1 ALERT# | GND | |
| 18 | NC | NC | NC | GND | NC | GND | |
| 17 | NC | GND | PRST# | REQ6# | GNT6# | GND | J |
| 16 | NC | NC | DEG# | GND | NC | GND | 2 |
| 15 | NC | GND | FAL# | REQ5# | GNT5# | GND | |
| 14 | AD[35] | AD[34] | AD[33] | GND | AD[32] | GND | C |
| 13 | AD[38] | GND | V(I/O) | AD[37] | AD[36] | GND | O |
| 12 | AD[42] | AD[41] | AD[40] | GND | AD[39] | GND | N |
| 11 | AD[45] | GND | V(I/O) | AD[44] | AD[43] | GND | N |
| 10 | AD[49] | AD[48] | AD[47] | GND | AD[46] | GND | E |
| 9 | AD[52] | GND | V(I/O) | AD[51] | AD[50] | GND | C |
| 8 | AD[56] | AD[55] | AD[54] | GND | AD[53] | GND | T |
| 7 | AD[59] | GND | V(I/O) | AD[58] | AD[57] | GND | O |
| 6 | AD[63] | AD[62] | AD[61] | GND | AD[60] | GND | R |
| 5 | C/BE[5]# | 64EN# | V(I/O) | C/BE[4]# | PAR64 | GND | |
| 4 | V(I/O) | NC | C/BE[7]# | GND | C/BE[6]# | GND | |
| 3 | CLK4 | GND | GNT3# | REQ4# | GNT4# | GND | |
| 2 | CLK2 | CLK3 | SYSEN# | GNT2# | REQ3# | GND | |
| 1 | CLK1 | GND | REQ1# | GNT1# | REQ2# | GND | |
| Pin | A | B | C | D | E | F | |

**Table A-3 CompactPCI J2 Interface Pin-outs**

### A.5.3 CompactPCI Interface (J3) Pin-outs

The CompactPCI interface I/O connector J3 consists of a 114-pin connector with pins assigned as follows:

| Pin | A | B | C | D | E | F | |
|---|---|---|---|---|---|---|---|
| 19 | GND | GND | GND | GND | GND | GND | |
| 18 | LPa_DA | LPa_DA# | GND | LPa_DC | LPa_DC# | GND | |
| 17 | LPa_DB | LPa_DB# | GND | LPa_DD | LPa_DD# | GND | Ethernet |
| 16 | LPb_DA | LPb_DA# | GND | LPb_DC | LPb_DC# | GND | |
| 15 | LPb_DB | LPb_DB# | GND | LPb_DD | LPb_DD# | GND | |
| 14 | +3.3V | +3.3V | +3.3V | +5V | +5V | GND | TM POWER |
| 13 | I/O 5 | I/O 4 | I/O 3 | I/O 2 | I/O 1 | GND | |
| 12 | I/O 10 | I/O 9 | I/O 8 | I/O 7 | I/O 6 | GND | |
| 11 | I/O 15 | I/O 14 | I/O 13 | I/O 12 | I/O 11 | GND | |
| 10 | I/O 20 | I/O 19 | I/O 18 | I/O 17 | I/O 16 | GND | |
| 9 | I/O 25 | I/O 24 | I/O 23 | I/O 22 | I/O 21 | GND | |
| 8 | I/O 30 | I/O 29 | I/O 28 | I/O 27 | I/O 26 | GND | |
| 7 | I/O 35 | I/O 34 | I/O 33 | I/O 32 | I/O 31 | GND | PMC 2 I/O |
| 6 | I/O 40 | I/O 39 | I/O 38 | I/O 37 | I/O 36 | GND | |
| 5 | I/O 45 | I/O 44 | I/O 43 | I/O 42 | I/O 41 | GND | |
| 4 | I/O 50 | I/O 49 | I/O 48 | I/O 47 | I/O 46 | GND | |
| 3 | I/O 55 | I/O 54 | I/O 53 | I/O 52 | I/O 51 | GND | |
| 2 | I/O 60 | I/O 59 | I/O 58 | I/O 57 | I/O 56 | GND | |
| 1 | SPKR Out | I/O 64 | I/O 63 | I/O 62 | I/O 61 | GND | |
| **Pin** | **A** | **B** | **C** | **D** | **E** | **F** | |

**Table A-4 CompactPCI J3 Interface Pin-outs**

### A.5.4 CompactPCI Interface (J5) Pin-outs

The CompactPCI interface I/O connector J5 consists of a 132-pin connector with pins assigned as follows.

| Pin | A | B | C | D | E | F | |
|---|---|---|---|---|---|---|---|
| 22 | I/O 5 | I/O 4 | I/O 3 | I/O 2 | I/O 1 | GND | |
| 21 | I/O 10 | I/O 9 | I/O 8 | I/O 7 | I/O 6 | GND | |
| 20 | I/O 15 | I/O 14 | I/O 13 | I/O 12 | I/O 11 | GND | |
| 19 | I/O 20 | I/O 19 | I/O 18 | I/O 17 | I/O 16 | GND | |
| 18 | I/O 25 | I/O 24 | I/O 23 | I/O 22 | I/O 21 | GND | |
| 17 | I/O 30 | I/O 29 | I/O 28 | I/O 27 | I/O 26 | GND | PMC1 |
| 16 | I/O 35 | I/O 34 | I/O 33 | I/O 32 | I/O 31 | GND | I/O |
| 15 | I/O 40 | I/O 39 | I/O 38 | I/O 37 | I/O 36 | GND | |
| 14 | I/O 45 | I/O 44 | I/O 43 | I/O 42 | I/O 41 | GND | |
| 13 | I/O 50 | I/O 49 | I/O 48 | I/O 47 | I/O 46 | GND | |
| 12 | I/O 55 | I/O 54 | I/O 53 | I/O 52 | I/O 51 | GND | |
| 11 | I/O 60 | I/O 59 | I/O 58 | I/O 57 | I/O 56 | GND | |
| 10 | GND | I/O 64 | I/O 63 | I/O 62 | I/O 61 | GND | |
| 9 | TR RST# | GND | GND | GND | GND | GND | |
| 8 | DD5 | DD9 | DD6 | DD8 | DD7 | GND | Primary |
| 7 | DD12 | DD3 | DD11 | DD4 | DD10 | GND | EIDE |
| 6 | DD0 | DD14 | DD1 | DD13 | DD2 | GND | & |
| 5 | IORDY | DIOR# | DIOW# | DRQ | DD15 | GND | MUX I/O |
| 4 | DA2 | DA0 | DA1 | INTRQ | DACK# | GND | |
| 3 | LDRQ1# | MUXIO_DATA | MUXIO_CLK | CS3# | CS1# | GND | |
| 2 | LAD3 | LAD2 | LAD1 | LAD0 | GND | GND | LPC Bus |
| 1 | USBD1# | USBD1 | SERIRQ | LFRAME# | TM_CLK | GND | & USB |
| Pin | A | B | C | D | E | F | |

**Table A-5 CompactPCI J5 Interface Pin-outs**

### A.5.5 On-Board Mass Storage Option Connector (P6) Pin-outs

| Pin No. | Signal Name | Pin No. | Signal Name |
|---|---|---|---|
| 1 | IDE_RST# | 2 | GND |
| 3 | SDD7 | 4 | SDD8 |
| 5 | SDD6 | 6 | SDD9 |
| 7 | SDD5 | 8 | SDD10 |
| 9 | SDD4 | 10 | SDD11 |
| 11 | SDD3 | 12 | SDD12 |
| 13 | SDD2 | 14 | SDD13 |
| 15 | SDD1 | 16 | SDD14 |
| 17 | SDD0 | 18 | SDD15 |
| 19 | GND | 20 | +3.3V |
| 21 | SDREQ | 22 | GND |
| 23 | SDIOW# | 24 | GND |
| 25 | SDIOR# | 26 | GND |
| 27 | SIORDY | 28 | NC |
| 29 | SDDACK# | 30 | GND |
| 31 | INT15# | 32 | NC |
| 33 | SDA1 | 34 | PDIAG |
| 35 | SDA0 | 36 | SDA2 |
| 37 | SDCS1# | 38 | SDCS3# |
| 39 | ACTIVITY# | 40 | GND |
| 41 | +5V | 42 | +5V MOTOR |
| 43 | GND | 44 | NC |

**Table A-6 On-Board Mass Storage Option Interface Pin-outs**

### A.5.6 PMC Site Connectors (J11 - J14 and J21 - J24) Pin-outs

Signal assignments on the PMC connectors are shown in Tables A-7, A-8, A-9 and A-10.

| Pin No. | Signal Name | Pin No. | Signal Name |
|---|---|---|---|
| 1 | - | 2 | -12V |
| 3 | GND | 4 | INTA# |
| 5 | INTB# | 6 | INTC# |
| 7 | BUSMODE#1 | 8 | +5V |
| 9 | INTD# | 10 | - |
| 11 | GND | 12 | +3.3V†† |
| 13 | CLK | 14 | GND |
| 15 | GND | 16 | GNT# |
| 17 | REQ# | 18 | +5V |
| 19 | V (I/O) | 20 | AD(31) |
| 21 | AD(28) | 22 | AD(27) |
| 23 | AD(25) | 24 | GND |
| 25 | GND | 26 | C/BE(3)# |
| 27 | AD(22) | 28 | AD(21) |
| 29 | AD(19) | 30 | +5V |
| 31 | V (I/O) | 32 | AD(17) |
| 33 | FRAME# | 34 | GND |
| 35 | GND | 36 | IRDY# |
| 37 | DEVSEL# | 38 | +5V |
| 39 | GND | 40 | LOCK# |
| 41 | SDONE#† | 42 | SBO#† |
| 43 | PAR | 44 | GND |
| 45 | V (I/O) | 46 | AD(15) |
| 47 | AD(12) | 48 | AD(11) |
| 49 | AD(09) | 50 | +5V |
| 51 | GND | 52 | C/BE(0)# |
| 53 | AD(06) | 54 | AD(05) |
| 55 | AD(04) | 56 | GND |
| 57 | V (I/O) | 58 | AD(03) |
| 59 | AD(02) | 60 | AD(01) |
| 61 | AD(00) | 62 | +5V |
| 63 | GND | 64 | REQ64# |

V(I/O) = 3.3V
# denotes active low, † pulled high via 2.7kOhm resistor,
†† pulled high via 10kOhm resistor.

**Table A-7 PMC J11 and J21 Connector Pin-outs**

| Pin No. | Signal Name | Pin No. | Signal Name |
|---|---|---|---|
| 1 | +12V | 2 | - |
| 3 | - | 4 | - |
| 5 | - | 6 | GND |
| 7 | GND | 8 | - |
| 9 | - | 10 | - |
| 11 | +3.3V†† | 12 | +3.3V |
| 13 | RST# | 14 | GND |
| 15 | +3.3V | 16 | GND |
| 17 | - | 18 | GND |
| 19 | AD(30) | 20 | AD(29) |
| 21 | GND | 22 | AD(26) |
| 23 | AD(24) | 24 | +3.3V |
| 25 | IDSEL | 26 | AD(23) |
| 27 | +3.3V | 28 | AD(20) |
| 29 | AD(18) | 30 | GND |
| 31 | AD(16) | 32 | C/BE(2)# |
| 33 | GND | 34 | - |
| 35 | TRDY# | 36 | +3.3V |
| 37 | GND | 38 | STOP# |
| 39 | PERR# | 40 | GND |
| 41 | +3.3V | 42 | SERR# |
| 43 | C/BE(1)# | 44 | GND |
| 45 | AD(14) | 46 | AD(13) |
| 47 | M66EN | 48 | AD(10) |
| 49 | AD(08) | 50 | +3.3V |
| 51 | AD(07) | 52 | - |
| 53 | +3.3V | 54 | - |
| 55 | PMC-RSVD | 56 | GND |
| 57 | PMC-RSVD | 58 | EREADY†† |
| 59 | GND | 60 | - |
| 61 | ACK64# | 62 | +3.3V |
| 63 | GND | 64 | +3.3V†† |

# denotes active low, † pulled high via 2.7kOhm resistor,
†† pulled high via 10kOhm resistor.

**Table A-8 PMC J12 and J22 Connector Pin-outs**

**NOTE:** Pins 58 and 64 are pulled high to suit Processor-PMC modules.

| Pin No. | Signal Name | Pin No. | Signal Name |
|---|---|---|---|
| 1 | - | 2 | GND |
| 3 | GND | 4 | C/BE(7)# |
| 5 | C/BE(6)# | 6 | C/BE(5)# |
| 7 | C/BE(4)# | 8 | GND |
| 9 | V(I/O) | 10 | PAR64 |
| 11 | AD(63) | 12 | AD(62) |
| 13 | AD(61) | 14 | GND |
| 15 | GND | 16 | AD(60) |
| 17 | AD(59) | 18 | AD(58) |
| 19 | AD(57) | 20 | GND |
| 21 | V(I/O) | 22 | AD(56) |
| 23 | AD(55) | 24 | AD(54) |
| 25 | AD(53) | 26 | GND |
| 27 | GND | 28 | AD(52) |
| 29 | AD(51) | 30 | AD(50) |
| 31 | AD(49) | 32 | GND |
| 33 | GND | 34 | AD(48) |
| 35 | AD(47) | 36 | AD(46) |
| 37 | AD(45) | 38 | GND |
| 39 | V(I/O) | 40 | AD(44) |
| 41 | AD(43) | 42 | AD(42) |
| 43 | AD(41) | 44 | GND |
| 45 | GND | 46 | AD(40) |
| 47 | AD(39) | 48 | AD(38) |
| 49 | AD(37) | 50 | GND |
| 51 | GND | 52 | AD(36) |
| 53 | AD(35) | 54 | AD(34) |
| 55 | AD(33) | 56 | GND |
| 57 | V(I/O) | 58 | AD(32) |
| 59 | - | 60 | - |
| 61 | - | 62 | GND |
| 63 | GND | 64 | - |
| V(I/O) = 3.3V | | | |

**Table A-9 PMC J13 and J23 Connector Pin-outs**

| Pin No. | Signal Name | Pin No. | Signal Name |
|---|---|---|---|
| 1 | I/O 1 | 2 | I/O 2 |
| 3 | I/O 3 | 4 | I/O 4 |
| 5 | I/O 5 | 6 | I/O 6 |
| 7 | I/O 7 | 8 | I/O 8 |
| 9 | I/O 9 | 10 | I/O 10 |
| 11 | I/O 11 | 12 | I/O 12 |
| 13 | I/O 13 | 14 | I/O 14 |
| 15 | I/O 15 | 16 | I/O 16 |
| 17 | I/O 17 | 18 | I/O 18 |
| 19 | I/O 19 | 20 | I/O 20 |
| 21 | I/O 21 | 22 | I/O 22 |
| 23 | I/O 23 | 24 | I/O 24 |
| 25 | I/O 25 | 26 | I/O 26 |
| 27 | I/O 27 | 28 | I/O 28 |
| 29 | I/O 29 | 30 | I/O 30 |
| 31 | I/O 31 | 32 | I/O 32 |
| 33 | I/O 33 | 34 | I/O 34 |
| 35 | I/O 35 | 36 | I/O 36 |
| 37 | I/O 37 | 38 | I/O 38 |
| 39 | I/O 39 | 40 | I/O 40 |
| 41 | I/O 41 | 42 | I/O 42 |
| 43 | I/O 43 | 44 | I/O 44 |
| 45 | I/O 45 | 46 | I/O 46 |
| 47 | I/O 47 | 48 | I/O 48 |
| 49 | I/O 49 | 50 | I/O 50 |
| 51 | I/O 51 | 52 | I/O 52 |
| 53 | I/O 53 | 54 | I/O 54 |
| 55 | I/O 55 | 56 | I/O 56 |
| 57 | I/O 57 | 58 | I/O 58 |
| 59 | I/O 59 | 60 | I/O 60 |
| 61 | I/O 61 | 62 | I/O 62 |
| 63 | I/O 63 | 64 | I/O 64 |

**Table A-10 PMC J14 and J24 Connector Pin-outs**

### A.5.7 Ethernet Connectors (P4 and P5) Pin-out

The front panel Ethernet Interfaces use 8-way RJ45 connectors with the following pin-out:

| Pin No. | Signal Name |
|---|---|
| 1 | DA |
| 2 | DA# |
| 3 | DB |
| 4 | DC |
| 5 | DC# |
| 6 | DB# |
| 7 | DD |
| 8 | DD# |

**Table A-11 Ethernet RJ-45 Connector Pin-outs**

### A.5.8 Port 80 (S2) Pin-outs

| Pin No. | Signal Name |
|---|---|
| 1 | GND |
| 2 | NC |
| 3 | Port 80 Select # |
| 4 | NC |
| 5 | D3 |
| 6 | D7 |
| 7 | D2 |
| 8 | D6 |
| 9 | D1 |
| 10 | D5 |
| 11 | D0 |
| 12 | D4 |
| 13 | +5 Volts |
| 14 | NC |
| # Denotes active low | |

**Table A-12 Port 80 Connector Pin-outs**

### A.5.9 USB Connector (P2 and P3) Pin-outs

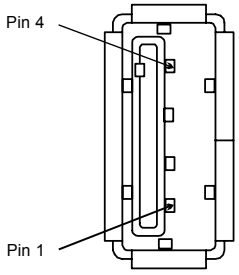



**Figure A-3 USB Connectors**

| Pin No. | Signal Name |
|---|---|
| 1 | +5V |
| 2 | USBD# |
| 3 | USBD |
| 4 | GND |

**Table A-13 USB Connector Pin-outs**

### A.5.10 Serial Port 10-way (LK4) IDC Connector

| Pin No. | Signal Name | Direction | Pin No. | Signal Name | Direction |
|---|---|---|---|---|---|
| 1 | DCD - Carrier Detect | Input to board | 10 | MAN_RST# | Input to board |
| 2 | DSR - Data Set Ready | Input to board | 9 | GND | - |
| 3 | RXD - Rx Data | Input to board | 8 | RI - Ring Indicator | Input to board |
| 4 | RTS - Request to Send | Output from board | 7 | DTR - Data Terminal Ready | Output from board |
| 5 | TXD - Tx Data | Output from board | 6 | CTS - Clear to Send | Input to board |

**Table A-14 COM1 LK4 Serial Port 10-way IDC Connector**

**WARNING:** If pin 10 is connected to GND then the board will reset.

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

## Gold-standard solutions

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

## We buy equipment

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

## Learn more!

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

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

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

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

ARTISAN® TECHNOLOGY GROUP