# Netra™ CP2060 and CP2080 Technical Reference and Installation Manual

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303-4900 U.S.A.
650-960-1300

Part No. 806-6658-10
July 2001, Revision A

Send comments about this document to: docfeedback@sun.com

Please
Recycle

Adobe PostScript

# Contents

# Figures

# Tables

# Preface

The Netra™ CP2060 and Netra™ CP2080 CompactPCI boards are a crucial building block that network equipment providers (NEPs) and carriers can use when scaling and improving the availability of next-generation, carrier-grade systems. Based on industry standards, it provides a low-cost solution easily customized to function as a system or satellite board. As a modular platform, the Netra CP2060 or the Netra CP2080 is easy to configure and install. It is designed to support requirements as a system controller board, for continuous uptime and scalability in the core network and access network infrastructures. It can also fit into any CompactPCI rack for use in a broad range of rack-mounted applications.

These boards incorporate UltraSPARC™-IIe technology microprocessors. They are highly integrated, high-performance CPU boards developed primarily for OEMs who want UltraSPARC performance with high availability for telecommunication applications. The Netra CP2060 and CP2080 boards are members of the CP2000 product family.

For the sake of convenience, this book will often refer to the Netra CP2060 and the Netra CP2080 boards as Netra CP2060/CP2080 boards when referring to both boards at the same time.

The *Netra CP2060 and CP2080 Technical Reference and Installation Manual* describes the hardware specifications, function and physical properties of these boards with a description of connector pinouts, boot sequence, diagnostics, installation and removal procedures.

## Who Should Use This Book

The *Netra CP2060 and CP2080 Technical Reference and Installation Manual* is written for our OEM customers, system integration engineers, field applications and service engineers, and others involved in the integration of these board into systems.

# How This Book Is Organized

**Chapter 1,** *Introduction* provides an overview of the Netra 2060 and Netra CP2080 boards.

**Chapter 2,** *Specifications* provides a summary of the specifications of the Netra CP2060 and CP2080.

**Chapter 3,** *Hardware and Functional Description* provides a description of the function of the various blocks on the CP2060/CP2080 boards.

Chapter 4, *Interrupts and Addresses* provides tables on CompactPCI interface, registers and interrupts.

**Chapter 5,** *Installation* describes some of the typical system configurations in which the Netra CP2060/CP2080 boards may be used and some typical installation procedures on the board.

**Chapter 6,** *Firmware* provides information on the Netra CP2060/CP2080 system OpenBoot firmware and Power-on Self Test (POST).

**Chapter 7,** *System Diagnostics—SunVTS* provides information on the comprehensive software package that tests and validates the Netra CP2060/CP2080 board.

**Chapter 8,** *Connectors, Pinouts and Switch Settings*, provides pinouts of the CompactPCI connectors on the Netra CP2060/CP2080 and some otherkey connectors on the boards.

**Chapter 9,** *Mechanical and Thermal Characteristics* provides mechanical illustrations and thermal characteristic details for the Netra CP2060/CP2080 boards.

*Bibliography* gives a list of general references and references to Sun documents.

*Glossary* lists definitions of many of the special terms used in this publication.

# What Typographic Changes Mean

The following table describes the typographic changes used in this book

**Table P-1** Typographic Conventions

| Typeface or Symbol | Meaning | Example |
|---|---|---|
| `AaBbCc123` | The names of commands, files, and directories; on-screen computer output | Edit your `.login` file.<br>Use `ls -a` to list all files.<br>`machine_name% You have mail.` |
| **`AaBbCc123`** | What you type, contrasted with on-screen computer output | `machine_name% su`<br>`Password:` |
| *AaBbCc123* | Command-line placeholder: replace with a real name or value | To delete a file, type `rm` *filename*. |
| *AaBbCc123* | Book titles, new words or terms, or words to be emphasized | Read Chapter 6 in *User's Guide.* These are called *class* options.<br>You *must* be root to do this. |

# Using UNIX Commands

To use UNIX® commands, please refer to the UNIX® manuals.

# Shell Prompts

| Shell | Prompt |
|---|---|
| C shell | *machine_name*% |
| C shell superuser | *machine_name*# |
| Bourne shell and Korn shell | $ |
| Bourne shell and Korn shell superuser | # |

# Accessing Sun Documentation Online

The `docs.sun.com`SM web site enables you to access a select group of Sun technical documentation on the Web. You can browse the `docs.sun.com` archive or search for a specific book title or subject at:

`http://docs.sun.com`

# Ordering Sun Documentation

Fatbrain.com, an Internet professional bookstore, stocks select product documentation from Sun Microsystems, Inc.

For a list of documents and how to order them, visit the Sun Documentation Center on Fatbrain.com at:

`http://www.fatbrain.com/documentation/sun`

# Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions. You can email your comments to Sun at:

`docfeedback@sun.com`

Please include the part number of your document in the subject line of your email.

CHAPTER 1

# Introduction

The Netra™ CP2060 and Netra™ CP2080 boards enable the user to integrate a telecommunication grade application. These boards are used for application processing in CompactPCI carrier-grade systems. The Netra CP2060/CP2080 board incorporates a 500-MHz UltraSPARC™ processor. It also works with a robust Solaris™ operating environment enhanced for advanced availability requirements. The Netra CP2060/CP2080 board supports advanced architectural capabilities for next-generation network infrastructure. The boards can function as satellite boards or as system host boards in a CompactPCI system.

These boards provide PICMG CompactPCI compliance (for details, see *Table 2-1*) with features to work in CompactPCI systems. In addition, with the introduction of PMC support, the Netra CP2060/CP2080 board enables OEM customers to mix and match third-party PMC cards, making it easier for them to tailor solutions to their specific application needs. An example of a typical Netra CP2080 board is illustrated in *Figure 1-1*.

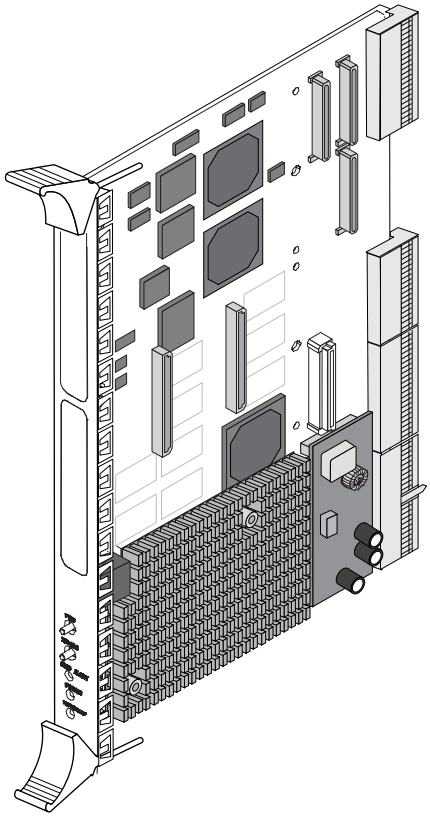


**Figure 1-1** A Netra CP2080 System Host Board or Satellite Board

# 1.1 Netra CP2060 / CP2080 System Configurations

Systems that conform to CompactPCI specifications require differentiation of chassis board slots depending upon the mode in which a board is to function.

To function as a system host board, a Netra CP2060 / CP2080 processor board has connections that include those that distribute PCI clocks and receive interrupts from peripherals. It must also be situated in a system slot in the CompactPCI segment because only these chassis slots have backplane wiring with the full set of

connections required to enable the system host board function. The Netra CP2060/CP2080 boards can control CompactPCI peripheral hardware on their own account—they assume a system controller role if they are installed into a PICMG system slot because they see a special enable signal (SYS-EN signal) in this slot. A system host slot is marked with an open triangle legend specified in *CompactPCI Specification*, PICMG 2.0 R3.0—see *Bibliography*.

The Netra CP2060/CP2080 board can be placed in one of the remaining (non-system controller) slots if the user wants to use it as a functional satellite board.

For a definition of the system host board and satellite board please see the *Glossary* in this book.

## 1.2 Properties of Netra CP2060/CP2080 CompactPCI Host Boards

The Netra CP2060/CP2080 boards are designed to support Hot Swap operation, system management, and environmental monitoring. These functions are implemented in the *System Management Controller* which is designed into the Netra CP2060/CP2080 boards. In addition, these products have PCI Mezzanine Connectors (PMC) that enable connection of compatible PCI peripherals.

*Table 1-1* shows some of the properties of the Netra CP2060 and CP2080 boards. There are two choices of board for the customer with regard to number of PMC ports and memory options. SDRAM on these two boards is mutually exclusive and configured in attachable module form or fixed on-board. Each of the two variants has two 10/100Mb ethernet ports, and two serial ports. In each case, two USB ports are taken to the CompactPCI backplane but only one is made available at the rear panel of the corresponding transition card.

**Table 1-1** Netra CP2060/CP2080 Product Comparison

| Board Product | Type | Processor | Memory | PMC slots | Hot Swap capability[1] |
|---|---|---|---|---|---|
| Netra CP2060 | Satellite Board and System Host Board capability | 500 MHz UltraSparc-IIe, integrated, 256KB, 4-way, set-associative, internal L2 cache in 2:2 mode | 512MB of fixed, on-board, unbuffered, 64-bit SDRAM with ECC (no stackable memory) | Slots for two PMC cards | As system host board: Basic and Full hot swap support<br><br>As satellite board: Basic and Full hot swap support |
| Netra CP2080 | Satellite Board and System Host Board capability | 500 MHz UltraSparc-IIe, integrated, 256KB, 4-way, set-associative, internal L2 cache in 2:2 mode | 256MB up to 2GB SDRAM in a stackable memory module (no soldered memory) | Slots for one PMC card (with up to 1 GB memory) No PMC cards if greater than 1GB memory utilized) | As system host board: Basic and Full hot swap support<br><br>As satellite board: Basic and Full hot swap support |

1. See *Table 2-1* for compliance.

For illustrations of the Netra CP2060/CP2080 board and to check for visual differences, please see *Figure 3-2* and *Figure 3-3*. For installation please refer to the Chapter 5, *Installation*.

## 1.3 Features

A summary of features of the Netra CP2060/CP2080 boards is given in *Table 1-2*.

**Table 1-2** Feature Summary

| Feature | Description |
|---|---|
| CPU | UltraSPARC IIe 500 MHz processor with internal 256KB L2 cache |
| Memory | Up to 2GB dependent upon model |
| Power requirement | Estimated at 17-18W (typical) and 23-25W (maximum) at 500MHz (excluding PMC power) |
| PICMG and PCI compliance | PICMG 2.0 R3.0 CompactPCI bus specification for 33MHz PCI speed<br>PICMG 2.1 R1.0 Hot Swap<br>PICMG 2.9 System Management<br>PCI revision 2.4 specifications |

**Table 1-2** Feature Summary

| Feature | Description |
|---|---|
| Host-mode support | Netra CP2060/CP2080 boards can function as a system host board |
| Satellite mode support | with Solaris 8 package |
| IPMI System management | uses IPMI communications with Baseboard Management Controller (BMC); performs Advanced System Monitoring (ASM) on local board interface for example temperature sense, FRU ID, and control |
| Hot Swap support | As system host board: Basic and Full hot swap support<br>As satellite board: Basic and Full hot swap support |
| Operating system | Solaris 8 operating environment, Release 1/01, or later compatible releases |
| Backplane IO[1]— accessible through rear transition card faceplate[2] | 2 Ethernet ports (10/100)<br>2 serial ports<br>1 USB port |
| PMC IO | Provision for adding up to two IHV supplied PMC expansion ports (2 on Netra CP2060, 1 on Netra CP2080) on front panel |
| Backplane PMC IO | 1 USB port; also provision for adding IHV-supplied PCI Interface Module (PIM) IO ports (2 on Netra CP2060 board, 1 on Netra CP2080 board) when used with transition card |
| Watchdog timer | 2-level watchdog timer |
| NVRAM | 8KB |
| System flash | 1MB on board |
| User flash | 4MB on board |
| Building compliance | NEBS Level 3 |
| Flash update | supported from downloaded file |

1. The user needs to use shielded cables for serial, Ethernet and USB ports on the transition card and the shield should be grounded at both ends.
2. The Netra CP2060/CP2080 board requires the use of a transition card ot be used in a cPCI system

# 1.4 Board Part Number, Serial Number and Revision Number Identification

The Netra CP2060/CP2080 board part number, serial number, and version can be found on stickers located on the card (see *Figure 1-2*). For proper identification of the board, please see the list below along with *Figure 1-2*.

The Sun barcode label provides the following information:

- Board part number (for example, 3753021) which is the first seven digits on the barcode label. The next six digits is the board serial number (for example, 005609). The boards may be identified by the following part numbers:

**Table 1-3** Identifying Netra CP2060 /CP2080 Boards by Board Part Number

| Board | Part number (first seven digits on label) |
|---|---|
| Netra CP2060 | 3753021 |
| Netra CP2080 | 3753023 |

The Subcon label provides the following information:

- Product part number (for example, 5762), product dash (for example, 02) and revision number (for example REV 57).
- Board date code (for example, 03/01) which represents the third week of year 2001.
- Country of origin (for example, ASSEMBLED IN CANADA)

---

**Note –** The CM barcode label in *Figure 1-2* is an optional label and may or may not be on the Netra CP2060/CP2080 board.

---

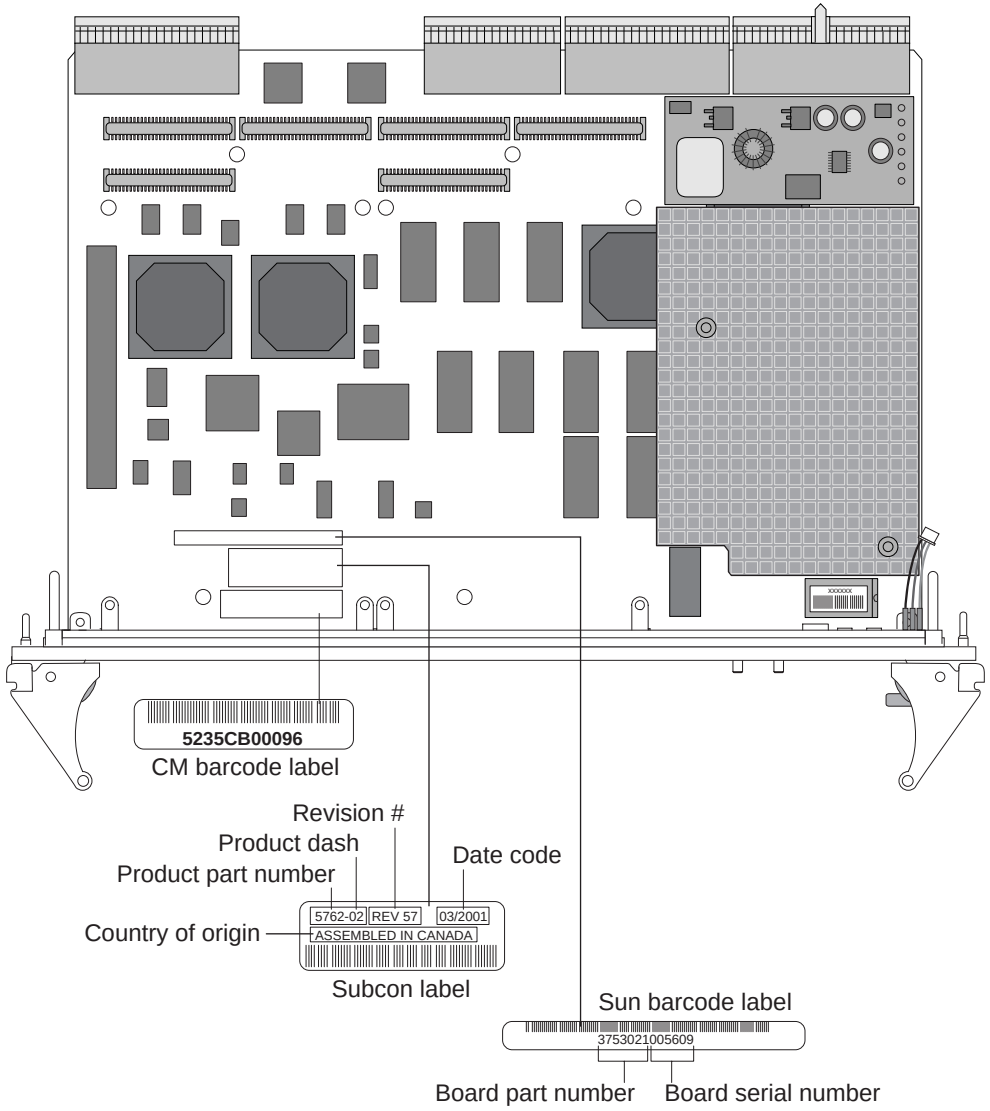



**Figure 1-2** Labeling on a Typical Netra CP2060 Board

---

**Note –** The board in *Figure 1-2* is a typical Netra CP2060 board. The labeling on the Netra CP2080 board is similar.

---

### 1.4.1 Firmware

See Section 6.7, *Determining Firmware Version*.

### 1.4.2 Solaris Software

These Netra CP2060/CP2080 boards require Solaris[SM] 8 operating environment, Release 1/01 software or later compatible releases.

## 1.5 Technical Support and Warranty

Should you have any technical questions or support issues that are not addressed in the Netra CP2060/CP2080 documentation set or on the Web site contact your local Sun Enterprise Services. This hardware carries a 1-year return-to-depot warranty. For customers in the US or Canada, please call 1-800-USA-4SUN (1-800-872-4786). For customers in the rest of the world, find the World Wide Solution Center nearest you by visiting our website:

*http://www.sun.com/service/contacting/solution.html*

When you call Sun Enterprise Services, be sure to indicate that the Netra CP2060/CP2080 was purchased separately and is not associated with a system. Please have the board identification information ready. The Netra CP2060/CP2080 board part number, serial number, and version can be found on stickers located on the board (see *Figure 1-2*). For proper identification of the board be prepared to give the representative the board part number, serial number, and date code (see Section 1.4, *Board Part Number, Serial Number and Revision Number Identification*).

CHAPTER 2

# Specifications

Specifications for the Netra CP2060 and Netra CP2080 are provided in the following tables.

## 2.1 Summary

**Table 2-1** Specifications Summary

| Property | Netra CP2060 | Netra CP2080 |
|---|---|---|
| CPU | 500 MHz UltraSparc-IIe | 500 MHz UltraSparc-IIe |
| Cache | Integrated, 256KB, internal 4 way set associative L2 cache operating in 2:2 mode | |
| Memory—SDRAM | 512MB on-board, unbuffered, 64-bit SDRAM with ECC | 256MB to 2GB SDRAM memory in a stackable module |
| Memory—boot flash | 1MB boot flash (firmware) on board | |
| Memory—user flash | 4MB on board | |
| CompactPCI specification compliance | PICMG 2.0 R3.0 CompactPCI bus specification for 33MHz PCI speed<br>PICMG 2.1 R1.0 Hot Swap compliance<br>PICMG 2.9 System Management | |
| Host modes support within system | Satellite board and System Host board modes | |
| System management | Capability provided for control by System Management Controller and BMC | |
| NVRAM | 8KB with replaceable battery | |

**Table 2-1** Specifications Summary

| Property | Netra CP2060 | Netra CP2080 |
|---|---|---|
| Firmware support | IPMI Protocol with System Management Controller (SMC);<br>Local Advanced System Monitoring (ASM);<br>Host CPU (UltraSPARC-IIe) communication;<br>Flash update execution from firmware;<br>Boot from user flash mode<br>Multiple PCI reset modes | |
| OS | Solaris 8 operating environment, Release 1/01 support or later compatible version | |
| Front Panel controls/ indicators | Reset (POR) and Abort (XIR) pushbuttons;<br>Hot Swap Blue LED, Alarm LED, Power LED;<br>2 PMC front-panel I/O cut-outs | |
| PMC card support | 2 PMC cards | 1 PMC card |
| Transition card | Connects IO on J3, J5 to rear panel; includes Ethernet, serial, and USB channels[1] | |
| Regulatory compliance | NEBS Level 3 Specification compliance (see Section 2.2.9, *Compliance*) | |
| OBP support | POST/OPB 4.0.xx | |

1. To meet emission standards, the user needs to use shielded cables for serial, Ethernet and USB ports on the transition card and the shield should be grounded at both ends.

# 2.2 Functional Specifications

This section provides various functional specifications.

## 2.2.1 System Compatibility

**Table 2-2** System Compatibility Attributes

| Property | Netra CP2060 | Netra CP2080 |
|---|---|---|
| Satellite board capability | yes | yes |
| System host capability | yes | yes |
| H110 chassis compatible | CompactPCI J4 is unconnected at board; enables this board to be used in H110 chassis | |
| NEBS compliance | NEBS Level 3 specification compliance | |
| CompactPCI compliance | PICMG 2.0 R3.0 CompactPCI bus specification for 33MHz PCI speed<br>PICMG 2.1 R1.0 Hot Swap specification<br>PICMG 2.9 System Management specification | |

## 2.2.2 CPU

**Table 2-3** CPU Specification

| Property | Netra CP2060 | Netra CP2080 |
|---|---|---|
| CPU | 500 MHz UltraSparc-IIe | |
| Mounting | 370-pin ceramic PGA package soldered to board | |
| Architecture | Sun 4U; SPARC V9 architecture with the VIS Instruction Set | |
| Cache | Integrated, 256KB, 4-way, set-associative internal L2 cache operating in 2:2 mode | |
| PCI bus local interface | PCI Bus 2.1 compatible, 33/66 MHz, 32-bit, 3.3V (internal to board only, does not come on connector) | |

## 2.2.3 Main Memory

**Table 2-4** Memory Specification

| Property | Netra CP2060 | Netra CP2080 |
|---|---|---|
| Memory size—min | 512MB | 256MB |
| Memory size—max. | 512MB | up to 2GB |
| Memory configuration | soldered on-board memory | no on-board soldered memory |
| Memory configuration | not expandable memory | up to two custom 198-pin stackable modules; see *Table 2-5* for allowable combinations |
| Memory type | 3.3V, synchronous DRAM with ECC LVTTL-compatible CMOS; configured on bus width of 64-bit + 8-ECC bits | |
| Identification to system | Serial EEPROM provides serial presence detect (SPD) to IPMI interface | |
| ECC | 8-bit; single bit error correction; double-bit error detection | |

### 2.2.3.1 Netra CP2060 Memory Configuration

The default memory configuration on the Netra CP2060 is 512MB. There is no stackable memory option on this board.

### 2.2.3.2 Netra CP2080 Memory Configuration

**Table 2-5** Memory Module Configurations Available on CP2080

| Bottom SDRAM Module P/N and Specification | Top SDRAM Module P/N and Specification | Total Memory Available on Board |
|---|---|---|
| 375-3024-01<br>256MB | none | 256MB |
| 375-3024-01<br>256MB | 375-3024-01<br>256MB | 512MB |
| 375-3025-01<br>512MB | none | 512MB |
| 375-3025-01<br>512MB | 375-3024-01<br>256MB | 768MB |
| 375-3025-01<br>512MB | 375-3025-01<br>512MB | 1024MB |
| 375-3026-01<br>1GB | none | 1GB |
| 375-3026-01<br>1GB | 375-3024-01<br>256MB | 1256MB |
| 375-3026-01<br>1GB | 375-3025-01<br>512MB | 1512MB |
| 375-3026-01<br>1GB | 375-3026-01<br>1GB | 2GB |

For directions on the installation process of the stackable memory module/s on the Netra CP2080, please refer to the document *Memory Module Installation/Removal Guide for CP2000 Family CompactPCI Boards* (P/N 816-0854-xx).

## 2.2.4 PCI Mezzanine Module (PMC) Interface

**Table 2-6** PMC Interface Specification

| Property | Netra CP2060 | Netra CP2080 |
|---|---|---|
| PMC module interfaces on system board | two: PMC A and PMC B | one: PMC A |
| Interface IEEE P1386.1 compliance | with draft 2.1 | |
| Connector configuration, PMC A (P1386 designations) | J21, J22 carry PCI signals; J24 module IO is connected to cPCI backplane J5; J23[1] | |

**Table 2-6** PMC Interface Specification

| Property | Netra CP2060 | Netra CP2080 |
|---|---|---|
| Connector configuration, PMC B (P1386 designations) | J11, J12 carry PCI signals; J14 module IO is connected to cPCI backplane J3; | not present |
| PMC connections to cPCI backplane | PMC A on J5; PMC B on J3 | PMC A on J5 |
| PCI clock | 33MHz | |
| PCI bus width | 32-bit | |
| Max power load -- per module, combined power rails (5V, 3.3V, 12V, -12V) | 7.5 W[2] | |

1. J23 is for internal Sun Microsystems use only and is not installed

## 2.2.5 Power Requirements

This section provides information on power sequencing and power requirements by connection phase.

### 2.2.5.1 Power Sequencing

**Table 2-7** Netra CP2080 Backplane Connector Power Requirements by Connection Phase

| Power Rail | Power Drawn from Backplane Connector at Phase: | | | | Description |
|---|---|---|---|---|---|
| | Early power on long pins[2] | Main power on medium pins[3] | | | |
| | Typ. (A) | Typ. (A/mA) | Max. (A) (with 256MB memory module) | Max (A) (with 2GB memory module)[4] | |
| +5V[1] | 0.35 | 2.20 | 3.0 | 3.0 | at connector J1/J2 |
| +3.3V | 0.02 | 2.20 | 2.5 | 3.5 | at connector J1/J2 |
| +12V | 0.00 | 40mA | 50mA | 50mA | at connector J1/J2 |
| –12V | 0.00 | 10mA | 20mA | 20mA | at connector J1/J2 |
| IPMB_PWR, | -- | -- | -- | -- | at J1/A4 |

1. The V I/O on the backplane is connected to 5V.
2. The typical figures provided for early power are only provided as examples
3. The typical figures are calculated as measured on the Netra CP2080 board, no PMC cards, with or without the XCP2060-TRN I/O Transition Card and while running SunVTS.

4. The maximum memory supported is up to 2GB.

**Table 2-8** Netra CP2060[1] Backplane Connector Power Requirements by Connection Phase

| Power Rail | Power Drawn from Backplane Connector at Phase: | | | Description |
|---|---|---|---|---|
| | Early power on long pins[3] | Main power on medium pins[4] | | |
| | Typ. (A) | Typ. (A) | Max. (A) | |
| +5V[2] | 0.35 | 2.20 | 2.7 | at connector J1 |
| +3.3V | 0.02 | 2.20 | 2.7 | at connector J1 |
| +12V | 0.00 | 40mA | 50mA | at connector J1 |
| –12V | 0.00 | 10mA | 20mA | at connector J1 |
| IPMB_PWR, | -- | -- | -- | at J1 / A4 |

1. The typical power requirement for the Netra CP2080 ranges from 2Amps to 2.2Amps depending on the stackable memory configuration range from 256Mb to 2GB.
2. The V I/O on the backplane is connected to 5V.
3. The typical figures provided for early power are only provided as examples.
4. The typical figures are calculated as measured on the Netra CP2080 board, no PMC cards, with or without the XCP2060-TRN I/O Transition Card and while running SunVTS.

## 2.2.6 Mechanical

These products comply with the mechanical specifications to be found in the CompactPCI specification PICMG 2.0 R3.0. See *Bibliography*.

## 2.2.7 Environmental Specifications

**Table 2-9** Environmental Conditions and Limits

| Ambient Conditions | Limits[1] | |
|---|---|---|
| Transportation and Storage Temperature | -40$^0$ C for 72 hrs. max. | +70$^0$ C for 72 hrs. max. |
| Transportation and Storage Humidity | 5% RH[2] non-condensing | 95% RH non-condensing |
| Operating Temperature | 0$^0$ C (-5$^0$ C short term) | 55$^0$ C (60$^0$ C short term) |
| Operating Humidity | 5% RH non-condensing | 85% RH (90% RH short term) non-condensing |
| Shock and Vibration | As stated in NEBS GR-630 CORE specifications, section 4.3.1 and 4.3.2 for shock criteria and 4.4.3 for vibration criteria; MIL-STD 810E, Method 514.4, CAT I MIL-STD 810E, Method 516.4, II-3.2 | |
| Electrostatic Discharge | NEBS GR-1089 Section 2 | |

1. Short term, in this column, refers to a period of not more than 96 consecutive hours and a total of not more than 15 days in 1 year.
2. RH is relative humidity.

## 2.2.8 Reliability / Availability

Reliability prediction is the first measurement point of expected behavior of the inherent design mean time between failures (MTBF) of the product. MTBF values calculated are shown in *Table 2-10*.

.

**Table 2-10** Reliability Prediction for Board Level MTBF

| Items | MTBF (hours) | Annualized Failure Rate (AFR in %)[2] |
|---|---|---|
| Netra CP2060 board[1] (includes 512MB soldered memory and power module) | 162,000 | 5.26% |
| Netra CP2080 board (includes 1 x 256MB memory cards and power module | 157,431 | 5.41% |
| XCP 2060-TRN card | 2,081,000 | 0.42% |
| Memory card 256 MB (P/N 375-3024-01) | 829,000 | 1.05% |
| Memory card 512MB (P/N 375-3025-01) | 778,000 | 1.12% |
| Memory card 1GB (P/N 375-3026-01) | 759,000 | 1.15% |

1. Board ambient temperature at $40^0$ C
2. AFR (%) is Annualized Failure Rate based on 8,760 power on hours (POH) per year.

## 2.2.9 Compliance

All printed wiring boards (PWBs) are manufactured by UL recognized manufacturers, with a flammability rating of 94-V0 or better. Compliance with EMI and safety regulations for products including the Netra CP2060/CP2080 board is entirely the responsibility of OEMs. The Netra CP2060/CP2080 board has passed FCC Class B tests in representative enclosures.

The Netra CP2060/CP2080 boards are intended to be incorporated into systems meeting the following regulations/compliances:

- USA FCC part 15 Class B
- USA Safety UL 1950
- Canadian ICES Class B
- Canadian Safety CSA C22.2 Number 950
- European Union EMC CE Mark EN55022 andEN50082-1
- European Union Safety CE Mark EN 60950
- European Union Safety TUV
- Japanese EMC VCCI Class B

- NEBS Level 3
  Board requirements for NEBS Level 3 criteria that provide the highest assurance of product operability with minimal service interruptions over the life of the equipment. It includes the following categories and all associated sections and subcategories:
  - GR-1089-CORE, Issue 2, Revision 1, February 1999 - Electromagnetic Compatibility and Electrical Safety - Generic Criteria for Network Telecommunications Equipment
  - NEBS GR-63-CORE, Issue 1, October 1995 - Network Equipment-Building System Requirements
  - NEBS GR-63-CORE, Issue 1, October 1995 - Network Equipment-Building System Requirements: Physical Protection

## 2.2.10 Safety

Please read the cautionary note provided below.

**Caution –** The Netra CP2060/CP2080 boards holds a lithium battery molded into the real-time clock, SGS # M4TXX-BR12SH (see *Figure 3-2* and *Figure 3-3* to determine the location of the real-time clock). Batteries are not customer replaceable parts. They may explode if mishandled. Do not dispose of the battery in fire. Do not disassemble it or attempt to recharged it.

CHAPTER 3

# Hardware and Functional Description

## 3.1 Summarized Physical Description

The Netra CP2060/CP2080 board is a 6U-sized CompactPCI circuit card with CompactPCI connectors J1 and J2 for PCI, and J3 and J5 for IO. J4 is not fitted to the board. See the illustrations *Figure 3-2* and *Figure 3-3*.

A Netra CP2060/CP2080 board is designed to function as a satellite board or as a system host board. The board has no integrated front-panel IO but it can provide IO either:

- through the CompactPCI backplane: a transition card is required to be connected to provide connector access for two serial ports, two Ethernet ports and a USB port. Sun Microsystems offers a compatible XCP2060-TRN Transition Card (P/N 375-0120-*xx*). This card is recommended for use, but an OEM customer may design their own transition card for use instead.
- through one or two PMC card interfaces provided to accept up to two IHV-supplied PMC IO cards. The number of cards that can be installed depends upon the particular model of board. See *Table 1-1* on page 26.
- through one or two IHV-supplied PIM modules which can duplicate front-panel PMC IO ports. The PIMs attach to interfaces on a transition card and bring the ports out to its panel at the rear of the enclosure. See Section 3.4.4, *PMC and PIM Interface* for further details.

Apart from ports of PMC modules that may be fitted into the Netra CP2060/CP2080 front-panel slots cut to accept them, the Netra board front panels include no connectors. The panel only includes status lamps and reset pushbuttons, listed top to bottom:

- ABORT —pushbutton (XIR)
- RESET — pushbutton (POR)
- ALARM/USER — red/green
- READY — green
- Blue LED for hotswap

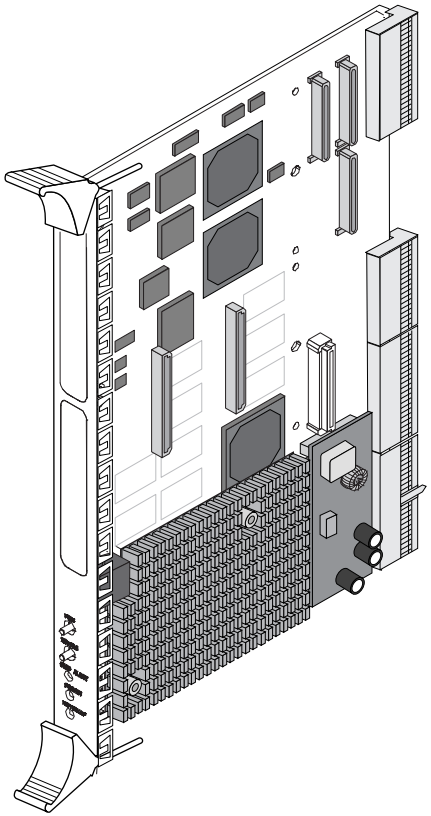


**Figure 3-1** Typical Netra CP2080 Board Assembly with Heatsink

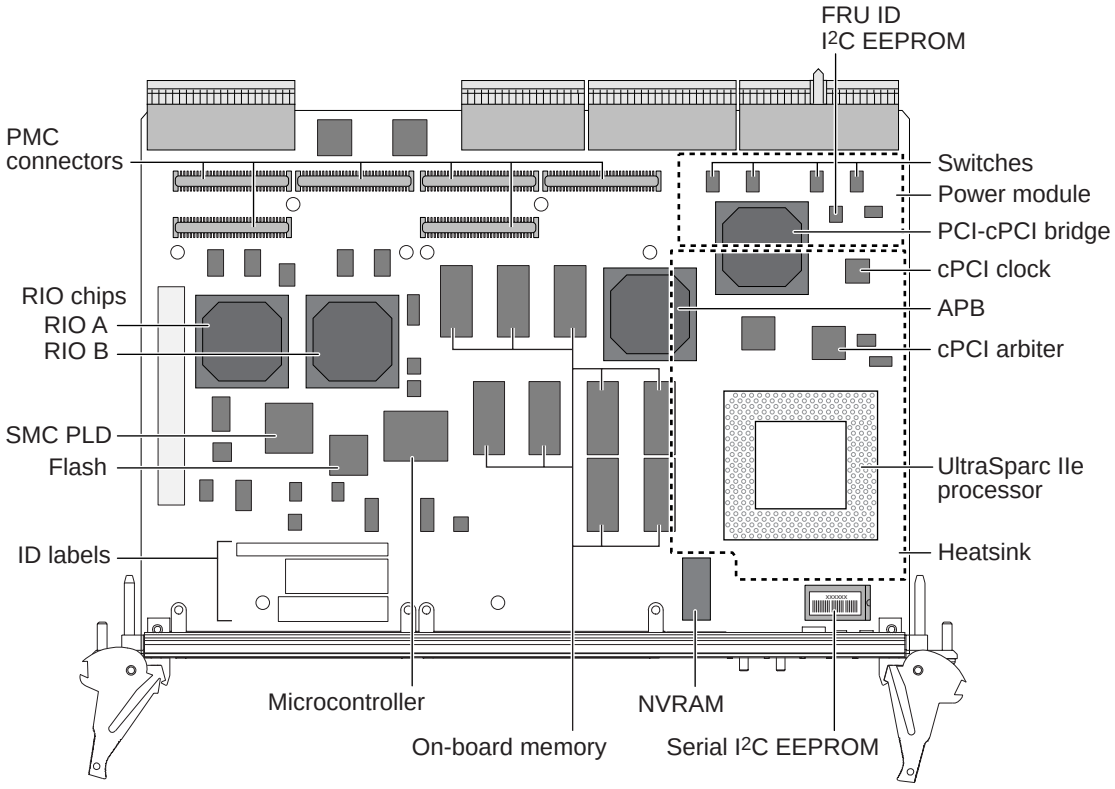



**Figure 3-2** Netra CP2060 Layout

---

**Note –** The heatsink and the power module are shown as dotted lines in this diagram to better illustrate the components on the board that lie beneath these devices.

---

For information on switches please see Chapter 8, *Connectors, Pinouts and Switch Settings.*

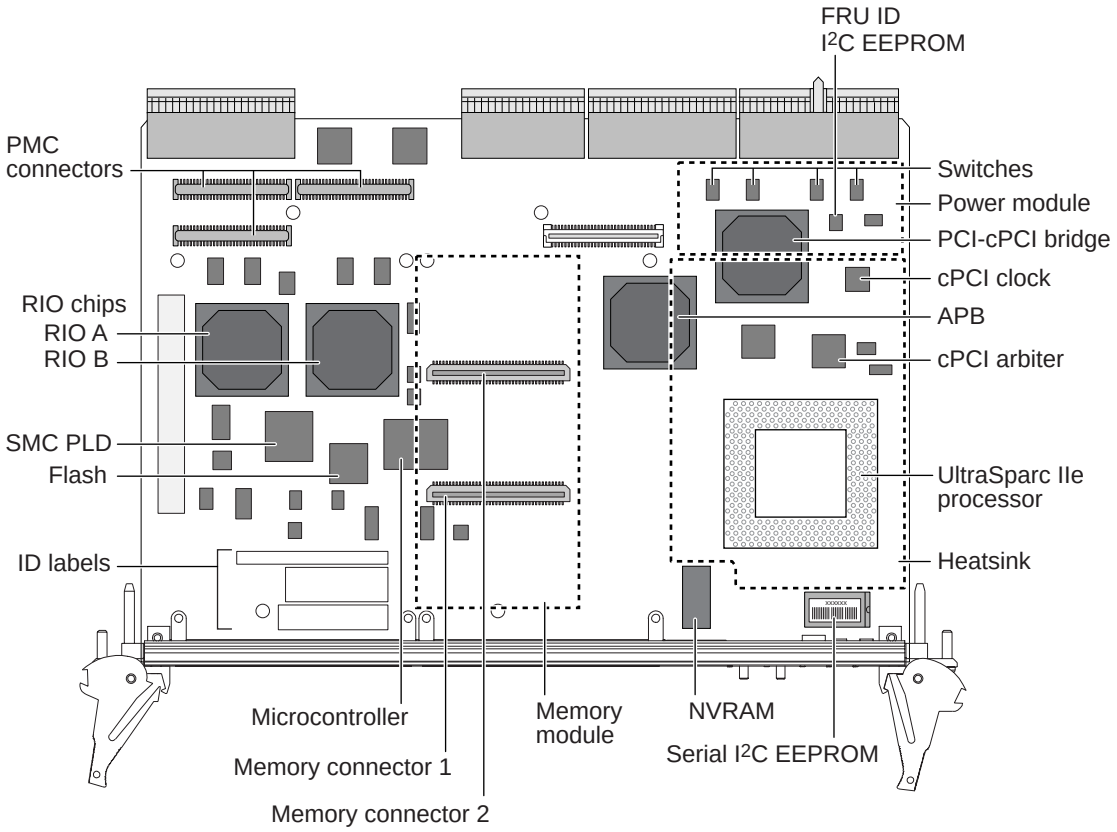



**Figure 3-3** Netra CP2080 Layout

---

**Note –** The heatsink, the power module and a stacked single-sized memory module are shown as dotted lines in this diagram to better illustrate the components on the board that lie beneath these devices.

---

For information on switches please see Chapter 8, *Connectors, Pinouts and Switch Settings*.

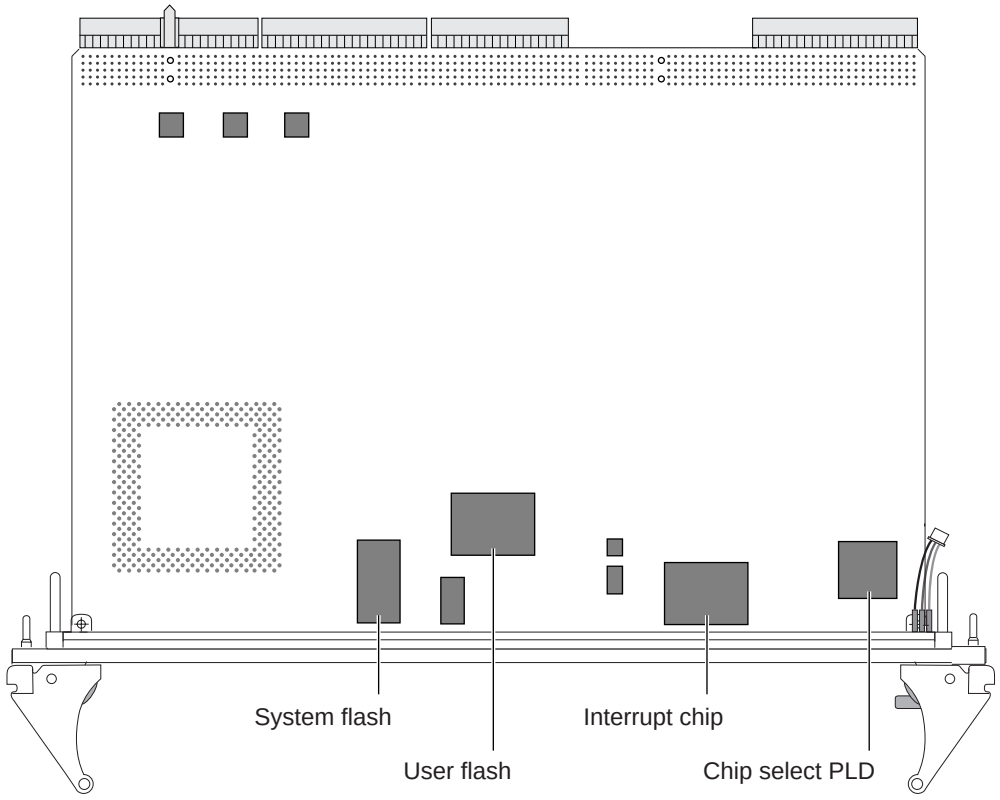



**Figure 3-4** A Typical Netra CP2060/CP2080 Board -- Solder Side

# 3.2 Detailed Description

A simplified schematic diagram is shown in *Figure 3-5*.

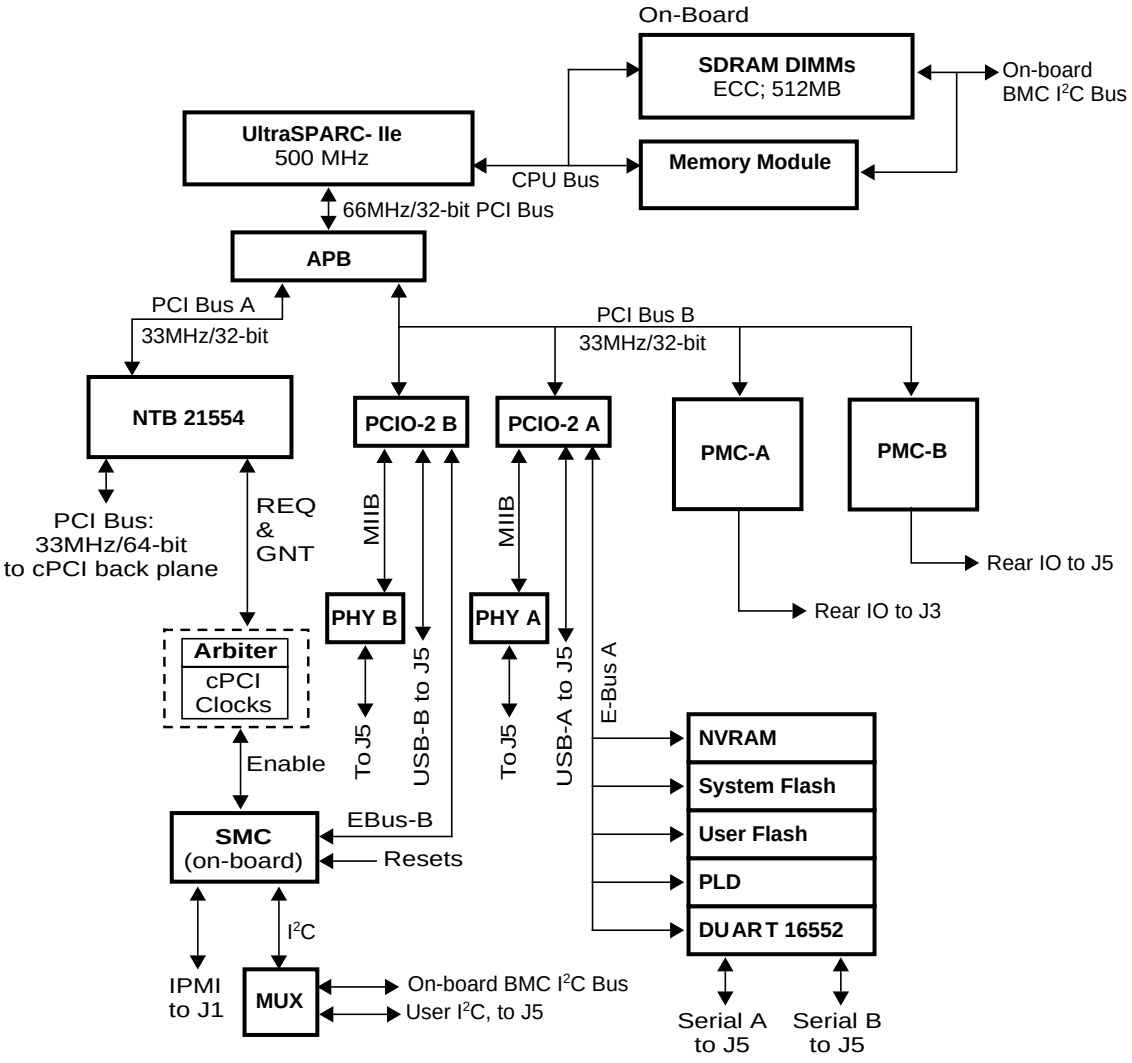



**Figure 3-5** Netra CP2060/CP2080 Functional Block Diagram

The UltraSPARC-IIe processor has its L2-cache integrated into its package. This processor is supported by SDRAM memory that may be soldered on to the board—in the case of the Netra CP2060 board, or available in plug-in module form on the Netra CP2080 board.

Apart from incoming interrupts, the processor handles all IO through its built-in 66MHz, 32-bit PCI bus interface. This interface is used to connect to a Sun Advanced PCI Bridge (APB) that services two 33MHz 32-bit downstream interfaces, PCI bus A, and PCI bus B.

PCI bus A connects to a non-transparent PCI bridge (21554 NTB) which services the principal PCI bus connection to the CompactPCI backplane through its connectors J1 and J2. In a system host role, a PCI bus arbiter provides CompactPCI bus arbitration signals for the CompactPCI backplane bus. It also supplies clocks for the CompactPCI bus. The arbiter is only active if the system host board is installed into the system slot and functions in a system host role. When the board is required to function as a satellite board, the CompactPCI bus arbiter is disabled by the System Management Controller (SMC)—see Section 3.4.3.1, *Arbitration in System Controller Role* for details.

PCI Bus B connects the APB to each of two PCIO-2 south bridge packages, PCIO-2 A and PCIO-2 B. See Section 3.4.2, *PCIO-2 Devices and E-bus Paths* for more details.

In addition, The PCI bus B from the APB connects to each of two 33MHz, 32-bit PMC interfaces on the host board. See Section 3.4.4, *PMC and PIM Interface* for more detail.

The SMC connects to:

- The bus arbiter and clock generator, enabling it to control CompactPCI bus arbitration, clock, and reset functions
- The on-board $I^2C$ bus, enabling it to communicate with sensors and controls
- The User IPMI bus, enabling user managements of other entities in the system. Peripheral Hot-swap control is also enabled through this path.
- The power module.

The SMC controls the startup of the board because it activates the power module and can control the system reset signals. In addition it handles Hot Swap signals from the CompactPCI backplane, for example: ENUM, HEALTHY, BD_SEL, and PCI_RST (the PCI reset signal). See Section 3.9, *Hot Swap* for more information on hot swap.

The block schematic diagram of *Figure 3-6* shows a more detailed block diagram of the Netra CP2060/CP2080 board. This figure contains blocks that are shown bordered by dashed lines. These blocks refer to configuration of:

- Main SDRAM memory in a fixed configuration with packages soldered to the system board on the Netra CP2060 board.
- Detachable memory modules are supplied for the Netra CP2080 only.
- PMC B is only supported on the Netra CP2060 board

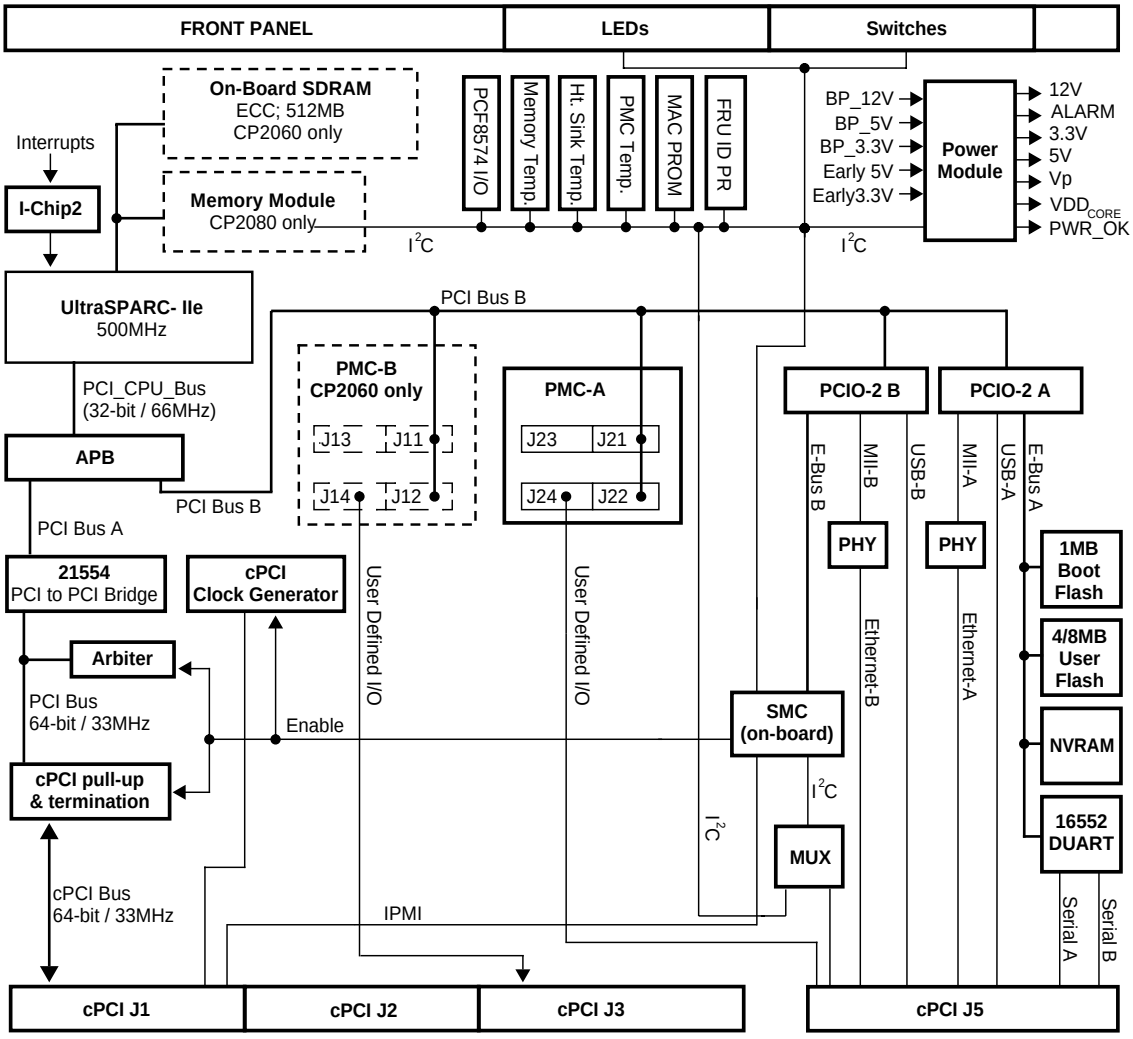



**Figure 3-6** Netra CP2060/CP2080 Detailed Block Diagram

---

**Note –** Dashed blocks show build variants.

---

# 3.3 CPU and Main Memory Subsystems

This section describes the UltraSPARC-IIe processor and additional memory on the Netra CP2060/CP2080 boards.

## 3.3.1 UltraSPARC-IIe Processor

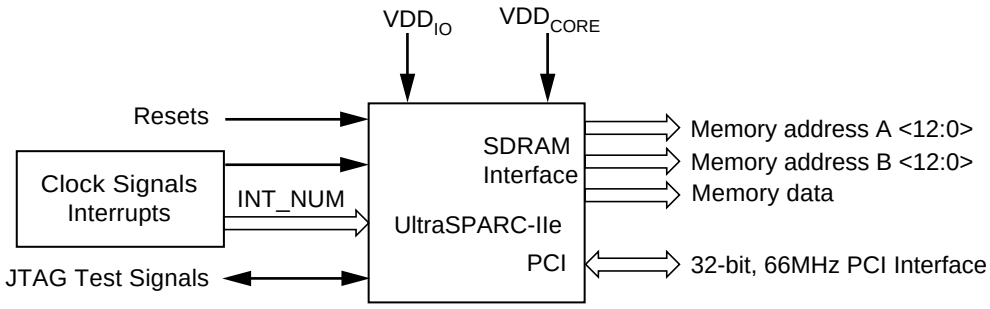



**Figure 3-7** UltraSPARC-IIe Interface

These Netra boards use the UltraSPARC-IIe 500MHz processor.

The processor is housed in a 370-pin ceramic Pin Grid Array (PGA) package. It typically dissipates no more 10 W at 500Mhz.

The UltraSPARC-IIe processor is directly connected to the board SDRAM through a 72-bit ECC path. Dual address buses are used to reduce capacitive loading and increase the memory density that can be accomplished with unbuffered devices.

The CPU connects to the APB by means of a 32-bit, 66 MHz PCI interface which the APB in turn translates to two downstream 33MHz PCI buses.

The UltraSPARC processor begins execution from a fixed image in a PROM that lies on E-bus A. The processor accesses this E-bus in a boot path that automatically includes the APB, PCIO-2 A (a South Bridge), and E-bus A.

Processor resets are received from the System Management Controller (SMC). See Section 3.7.1, *Resets* and *Figure 3-22* for more detail.

The various interrupts on the board are prioritized and encoded by the I-chip2 to appear at the UltraSPARC-IIe processor as 6-bit parallel data. See Section 3.7.2, *Interrupts* for more information.

The processor IO is run at a fixed VDDIO of 3.3V but the core voltage, VDDCORE, is adjustable and configured according to CPU speed, typically in the range of 1.3 V to 1.9 V.

JTAG / Test signals are available for use in boundary scan diagnostics.

## 3.3.2 Memory Address Mapping

The UltraSPARC-IIe L2 cache megacell reserves a 2 GB region for cacheable main memory. In the case of the CP2080 board, the memory databus width and the module databus width are of equal size (64 bit data plus 8 bit ECC) so memory modules can be installed in mixed sizes.

The UltraSPARC-IIe Address Data Generation Logic (ADGL) logically maps modules according to their size, rather than their physical location. The largest sizes are mapped to the lowest address ranges. Where modules of identical size are present, the lower slot number is mapped to the lower address range.

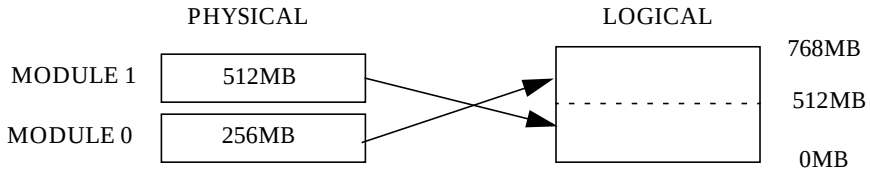



**Figure 3-8** Memory Mapping Example

## 3.3.3 SDRAM Memory

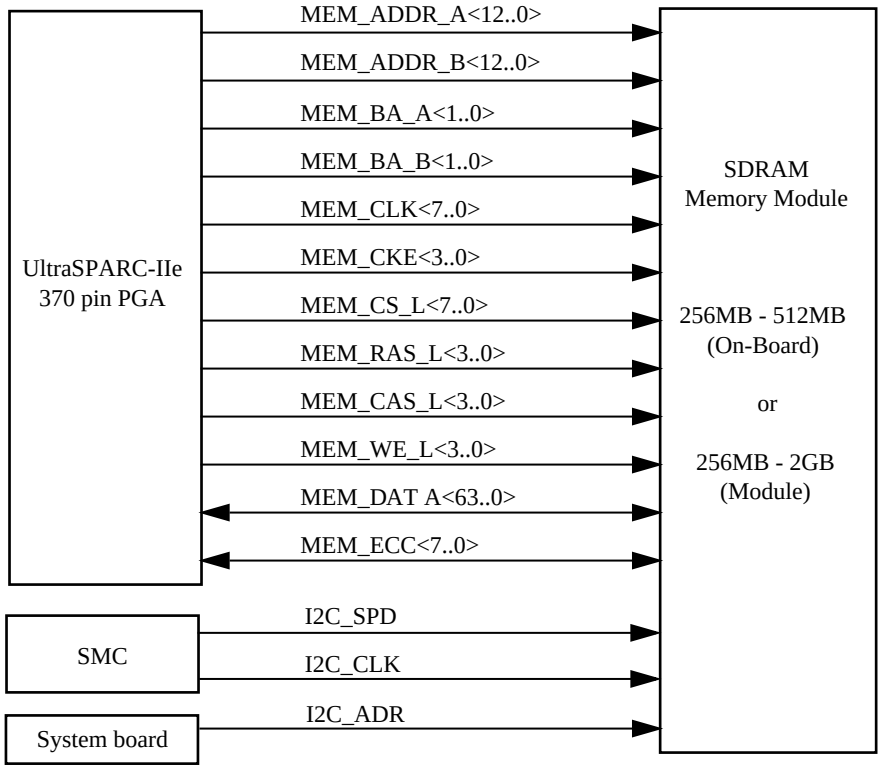



**Figure 3-9** SDRAM Memory Interface

The UltraSPARC-IIe 500MHz processor connects directly to the memory with a 72 bit ECC data bus. The memory can be either on-board SDRAM or composed of up to two mezzanine memory modules, dependent upon the particular board.

Each mezzanine memory module has two 100-pin male connectors on its bottom surface—these plug into corresponding female connectors on the system board. The module also has two of the same type of female connectors on its top side to enable the addition of an upper module. No more than two modules may be stacked.

Each module is equipped with a temperature sensor and a serial EEPROM device containing 256 bytes of presence detect data. The double size module (if used) also contains a PLL, which generates eight clock signals. Both the temperature sensor and the serial EEPROM are accessed using the two pin $I^2C$ protocol.

The memory block, whether on-board or a module, provides non-volatile serial memory to enable the Serial Presence Detect function. This memory can be interrogated by the SMC through the $I^2C$ interface.

## 3.3.4 Memory Components

This section describes additional memory that are available on the Netra CP2060/CP2080 boards.

### 3.3.4.1 System (Boot) Flash Memory

The system flash resides in 1MB of space. It contains Common Operations & Reset Environment (CORE) firmware, Comprehensive POST, and OBP boot code. The system flash may be upgraded by running a program out of OBP or executing a Solaris™ software script. If the system flash should become corrupted for any reason, you should contact your nearest Field Application Engineer.

### 3.3.4.2 User Flash Memory

The board is equipped with 4MB of user flash memory. The customer may use the flash memory for various purposes such as storage for RTOS, user data storage, OBP information or to house *dropins.* Dropins simplify customizing a system for the user.

A userflash switch SW2501 determines whether the userflash is detected during OBP boot and whether or not it is write enabled (please see Section 8.3, *Switch Settings*).

### 3.3.4.3 NVRAM

These boards use an 8K-bit X 8 timekeeper SRAM (NVRAM) package. This component provides:

- Battery backup using a removeable lithium battery
- A time-of-day (TOD) real-time clock
- 8KB storage for environment variables, user modifiable. The ethernet address and Host ID is stored in the NVRAM.
- On firmware boot up, the ethernet address stored in the NVRAM is compared against the backup copy stored in the Serial $I^2C$ EEPROM on the Netra CP2060/CP2080 board and the NVRAM is updated if it differs from the backup copy.

### 3.3.4.4 Serial $I^2C$ EEPROM

This device is also called the MAC address carrier and it stores the backup copy of the board MAC address and Host ID in a removable serial EEPROM that is accessible through the $I^2C$ bus. This data is downloaded to the NVRAM at the OBP level.

### 3.3.4.5 FRU ID $I^2C$ EEPROM

The FRU ID $I^2C$ EEPROM chip stores manufacturing related information of the CP2060/CP2080 board which is for internal Sun Microsystems service use only. This information is useful when the Netra board is being serviced and is not of any use to the OEM customer.

## 3.4 Bus Subsystems

There are four PCI buses on the board, three internal ones and the external CompactPCI bus that is driven to and from the backplane. One of the internal PCI buses, PCI bus B, is bridged to two lower-speed buses E-bus A and E-bus B. PCI bus A communicates with the CompactPCI backplane through the non-transparent PCI-to-PCI bridge (21554 NTB). This arrangement is shown in *Figure 3-10*.

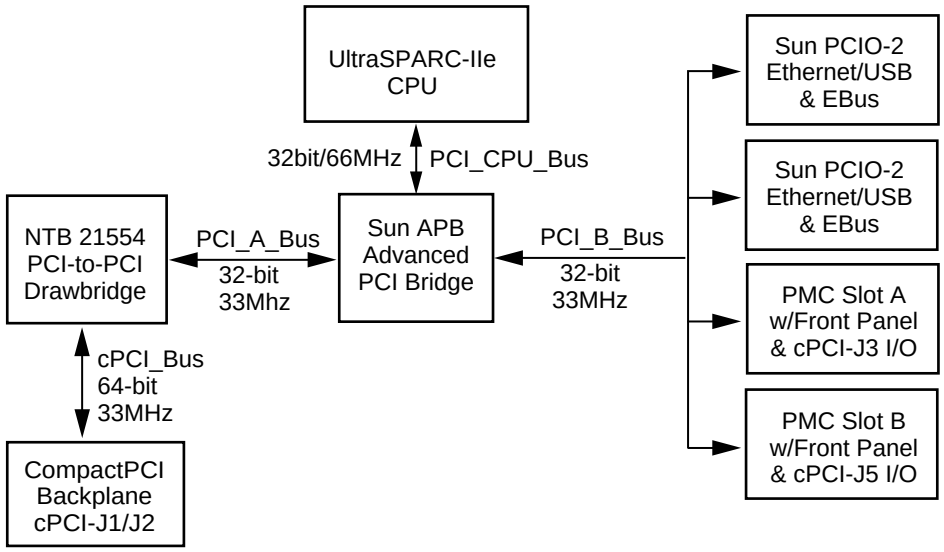



**Figure 3-10** Netra CP2060/CP2080 PCI Bus interface, 33MHz CompactPCI bridge

## 3.4.1 APB PCI Bus Interfaces

The UltraSPARC-IIe CPU has an integrated 32bit/66MHz PCI bus interface. The Advanced PCI Bridge—acting as a North Bridge—splits this bus into two 32-bit/33MHz PCI buses. Of these, the PCI A bus connects to the PCI non-transparent bridge which forms the interface to the CompactPCI backplane. The PCI B bus connects to two PCIO-2 bridges. Each of these bridges carry an E-bus and peripheral interfaces at their other end.

## 3.4.2 PCIO-2 Devices and E-bus Paths

The two PCIO-2 bridges connect between the APB PCI Bus B and their E-Bus and peripheral interfaces at their other end. Each of these bridges carry one E-Bus interface. It is used to interface slower internal peripherals.

E-Bus A coming from PCIO-2 A :

- System Flash EEPROM. The UltraSPARC-IIe processor accesses System Flash EPROM through the APB and PCIO-2 A to boot from location 0xF0000.0000.
- User Flash EPROM
- NVRAM

- Main PLD which provides E-bus decodes for chip select and CompactPCI arbiter control logic.
- Serial IO through a dual universal asynchronous receiver/transmitter (DUART) device which uses an RS232 transceiver device to support two independent RS232 style serial ports. The RS232 level signals feed to the XCP2060-TRN transition module through the cPCI J5 connector.

E-bus B coming from PCIO-2 B :

- The System Management Controller (SMC). This path is the primary means of communication between the UltraSPARC-IIe host and the SMC.

In addition, PCIO-2 A supports the MII Ethernet A port and USB A port. PCIO-2 B supports the MII Ethernet B port and USB B port.

## 3.4.3 CompactPCI Bus

The non-transparent bridge (NTB), in this case the Intel 21554 device, connects the 32-bit/33MHz internal PCI bus A to the 64-bit/33MHz CompactPCI backplane bus through CompactPCI connectors J1 and J2. This interface conforms to the PICMG 2.0 R3.0 CompactPCI Specification.

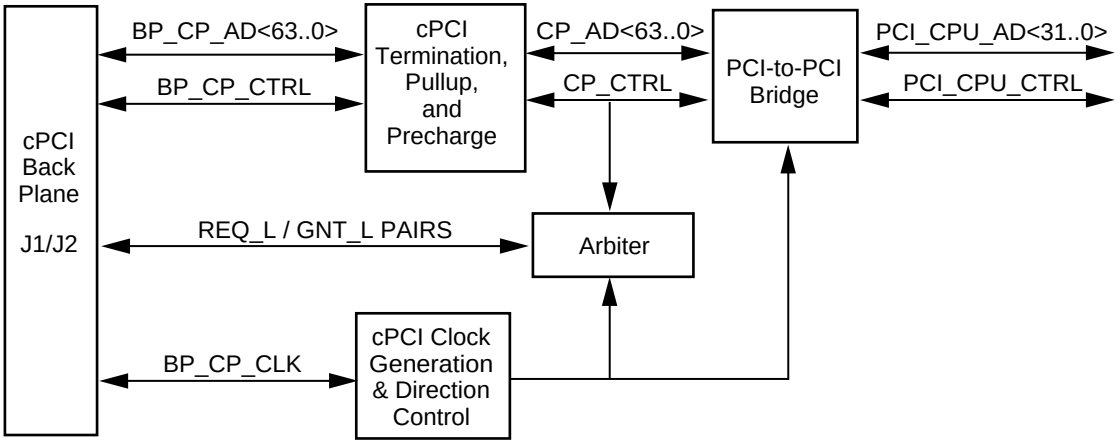



**Figure 3-11** CompactPCI Bus Interface

The arbiter—only used when the board functions in a system host board role—provides for orderly sharing of the CompactPCI bus among potential bus masters, or *initiators*. When the board performs in a system host board role, PCI clocks sourced from a clock generator on the board are driven to a CompactPCI CLK bus signal to all slots on the CompactPCI backplane.

### 3.4.3.1 Arbitration in System Controller Role

The Netra CP2060/CP2080 board may be used as a system controller board or as a satellite board. When the Netra board is a system controller board, the arbiter is enabled by the SMC. When the Netra board is a satellite board, the arbiter is disabled by the SMC through a control signal (signal SYS-EN on J2/C2).

*Figure 3-12* and *Figure 3-13* illustrate the signal flows for the two board operations.

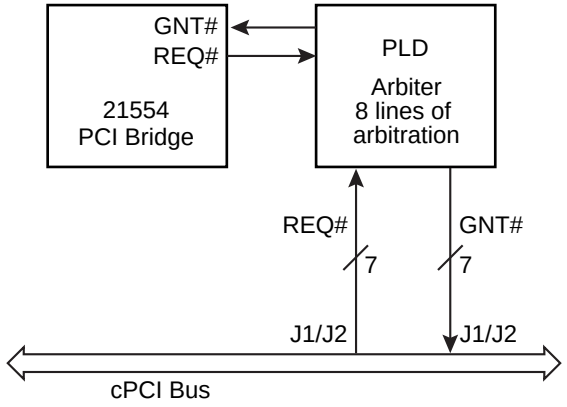



**Figure 3-12** Netra CP2060/CP2080 System Controller Board: REQ#/GNT# Signal Flow

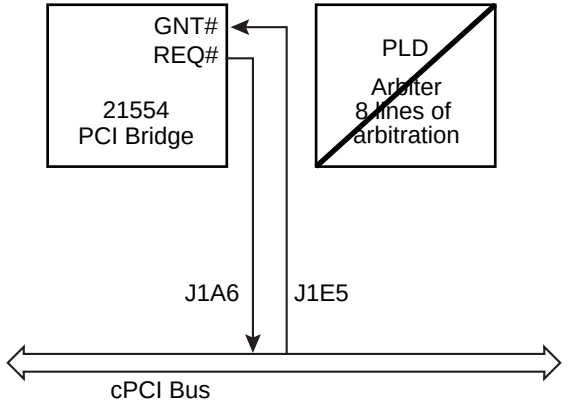



**Figure 3-13** Netra CP2060/CP2080 Satellite Board: REQ#/GNT# Signal Flow

When the Netra CP2060/CP2080 board is operating as a satellite board, the arbiter is disabled (through a control signal from the SMC module). When disabled, the arbiter tristates REQ1# through REQ6# and GNT1# through GNT6# and pulls them to a known state. The multiplexing, switching, and/or logic used to control the flow of the arbiter signals comply with all requirements of the cPCI specification, notably the requirements for single loads and stub length.

## 3.4.4 PMC and PIM Interface

The PCI Mezzanine Card (PMC) interface is defined by IEEE and PICMG standards:

- IEEE P1386, presently in draft form, defines the Common Mezzanine Card (CMC) mechanical profile (*common* because this definition, for example, can also apply to a VME rendering)
- P1386.1 defines the PCI electrical interface through its connectors Pn1/Jn1 through Pn4/Jn4
- PICMG 2.3 R1.0, *PMC on CompactPCI Specification* maps the PMC signals from the Jn1 through Jn4 connectors on the CompactPCI board through the CompactPCI backplane connections. In the case of the 6U products discussed here, these connections are routed to CompactPCI J3 and J5 connectors.

The purpose of the PMC interface is to enable the user to use Independent Hardware Vendor (IHV) PMC cards to implement a particular IO interface choice from the host at the system integration level. Cards in PMC slots are the only means of providing IO from the front-panels of Netra CP2060/CP2080 boards.

A matching Sun XCP2060-TRN IO transition card (see reference to the manual in *Bibliography*) fitted at the rear of the backplane provides for the attachment of the corresponding IHV-supplied PIM card hardware which comprises a PMC module—which may or may not carry an IO connector on its front flange—and a PIM card to connect to the transition card PIM sockets. These items together can duplicate the PMC front-panel interface at the transition card panel which is at the rear of the enclosure. Examples of such interfaces are display controller, Ethernet, SCSI, or T1 or T3 communications channels.

Using a PIM requires dedicating the corresponding PMC slot because the PMC board performs an adapter function between the PCI bus B and the user IO signals passed through J5 to the PIM slot on the transition card. When a PMC card that has a front-panel connector is used in this mode, jumpers are typically set to disable its front IO operation. See *Figure 3-14* for an illustration of the interconnection between PMC and PIM slots for installed PIMs. If PIM cards are used on the transition card, please note that PMC connectors and its corresponding PIM connectors should be used.

---

**Note –** The Netra CP2080 board only supports one PMC/PIM slot.

---

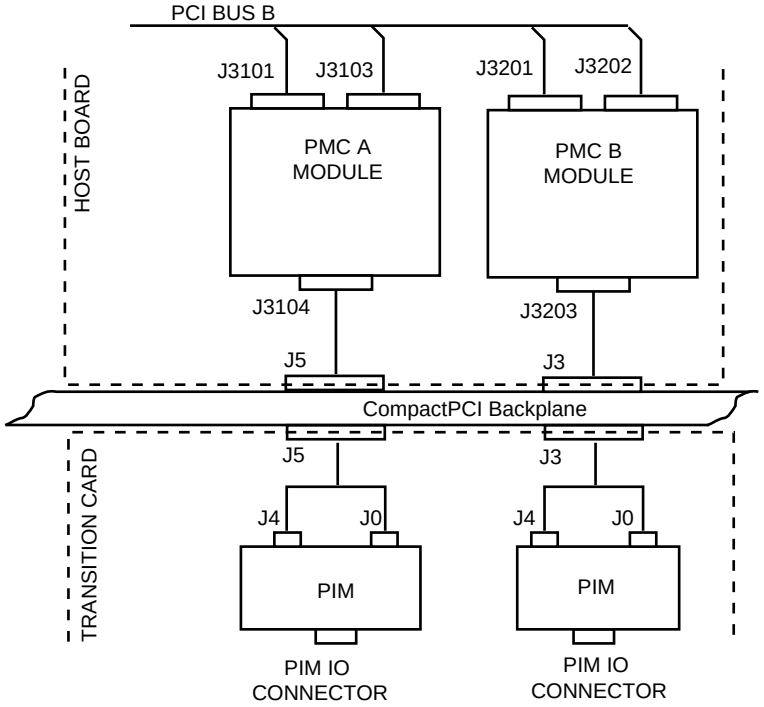



**Figure 3-14** PIM Installation Configuration

*Figure 3-15* shows the PMC-A and card attachment to a Netra CP2060/CP2080 board. A Netra CP2060 board has provision for the attachment of a second card, PMC-B, adjacent to the first—see *Figure 3-16*.

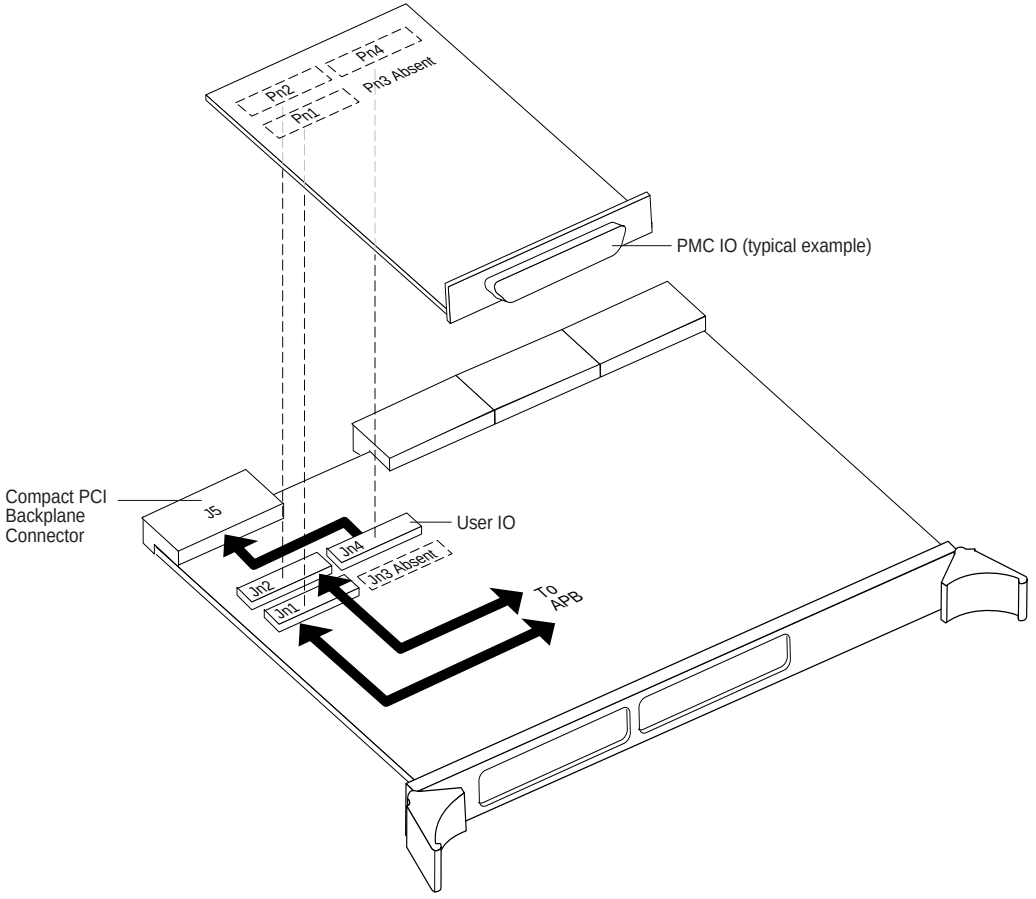



**Figure 3-15** Data Paths in PCI Mezzanine Module Interface on Host Board

The APB on the Netra CP2060/CP2080 board supplies PCI bus signals to PMC connectors J21 and J22. The PMC card logic decodes its specific IO interface which it makes available at the front panel.

64-pin PMC connectors J23 and J13 are not fitted to these Netra boards because this connector is specified for expansion for 64-bit PCI (it carries the upper 32 bits) which is not provided. A 64-bit capable PMC card can function in these slots but its bus interface is constrained to 32 bits. A special 80-pin connector can be fitted in the J3204 location of the Netra CP2080 board (see *Figure 3-17*). This connector is an IO connector used by Sun Microsystems™ only.

J24 is specified for user IO and carries PMC signals to CompactPCI backplane connector J3. If a transition card is installed, this J5 IO is conditioned by an IHV-supplied PIM to provide matching IO on the enclosure back panel. Its backplane IO is routed to CompactPCI/J5 connector.

In the case of the PMC B card for Netra CP2060, J11 and J12 are similarly connected to the APB but the user IO from J14 is routed out of CompactPCI connector J3.

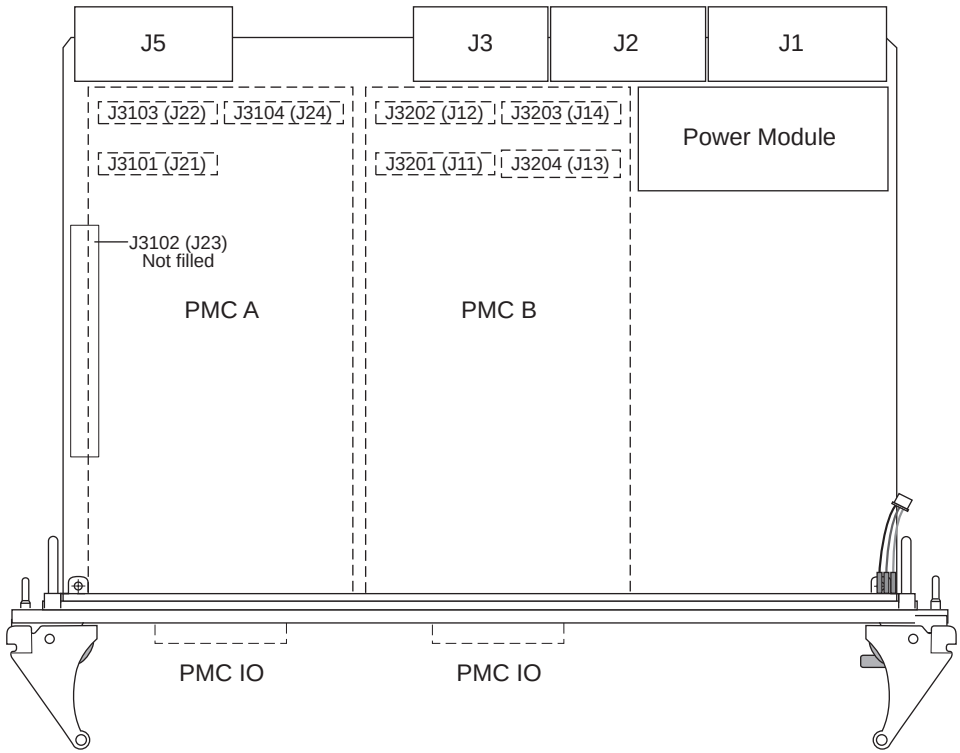



**Figure 3-16** PMC Connector Interfaces on Netra CP2060

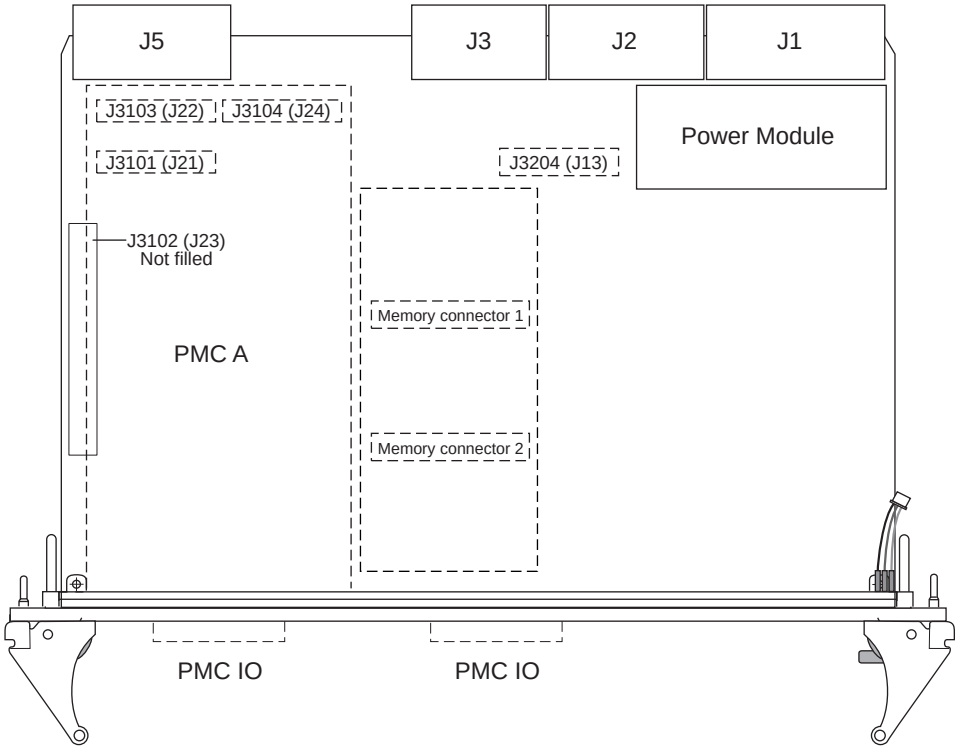



**Figure 3-17** PMC Connector Interface and Typical Stacked Memory Module on Netra CP2080

## 3.4.5 $I^2C$ and IPMI Channels

The $I^2C$ paths are shown in *Figure 3-18*. $I^2C$ communication is used:

- To implement the IPMI interface for system management
- To provide a means of performing Local Advanced System Monitoring (ASM) which monitors—and controls where appropriate—local board or chassis "housekeeping" functions, which include: monitoring of temperature, FRU ID, MAC address, and memory type and size.
- To provide an interface for user monitoring and control, for example chassis temperature or fan operation

Each $I^2C$ device on the board uses common address pins. The devices are distinguished by the internal device ID. All $I^2C$ devices are supplied from early power before backend power is established (see Section 3.8, *Power Subsystem* for further details).

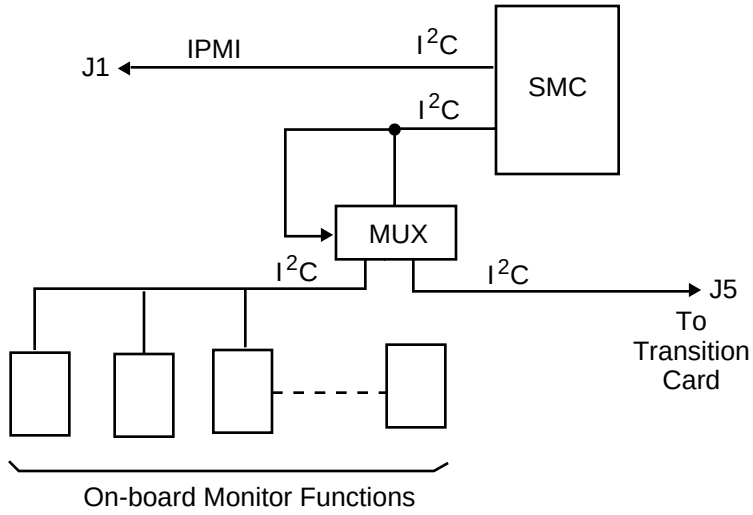



**Figure 3-18** Netra CP2060 and CP2080 I²C Paths

## 3.5 System Input/Output

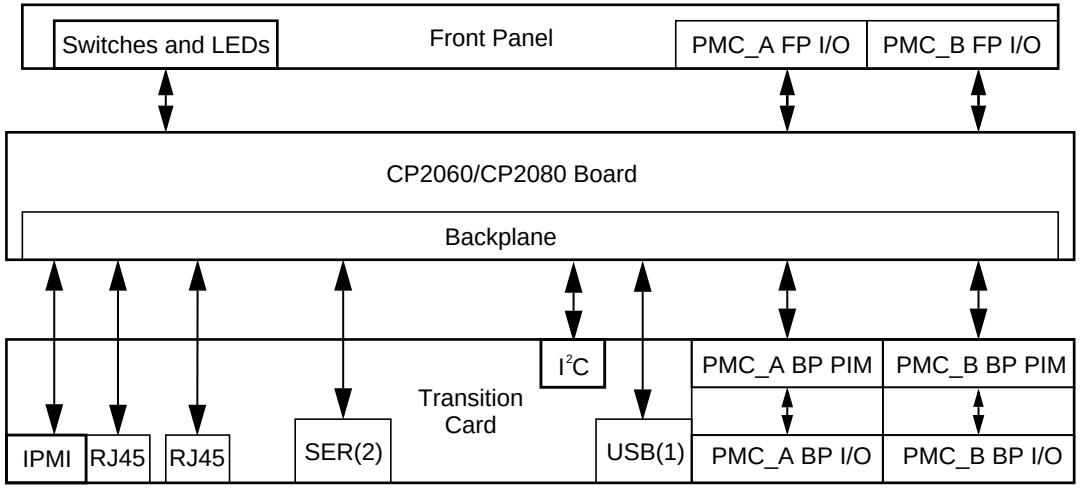



**Figure 3-19** I/O Interfaces

IO for the Netra board can be considered in four groups which are described in the following sections.

## 3.5.1 Front-panel IO

*Figure 3-20* illustrates the indicators and IO connectors on the Netra CP2060/CP2080 board front panel. The Netra CP2060/CP2080 board front panel connectors, buttons and LEDs are described below.

- Two peripheral mezzanine card (PMC) I/O bezels
- ABORT: An abort pushbutton switch; passes an XIR signal to the SMC
- RESET: A reset pushbutton switch; passes an Power-on-Reset (POR) signal to the SMC
- ALARM/USER: A red/green (two color) LED—red for board status, communicated from the SMC—green to indicate a user defined function programmed at the UltraSPARC host level. For details on using SMC commands please go to the product website at:

*http://www.sun.com/microelectronics/commprovider/cp2060/*

- READY: A green power LED, sourced from the power module
- A blue LED for hot-swap status, sourced from the SMC

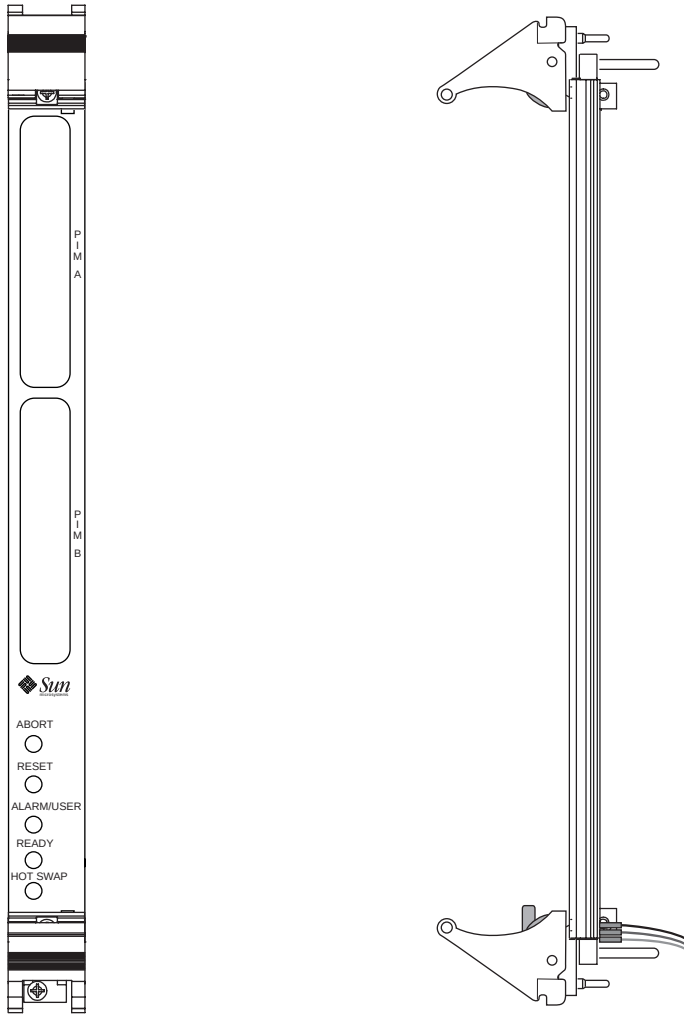



**Figure 3-20** Netra CP2060/CP2080 Board Front Panel

## 3.5.2 PMC Interface

The host board includes two PMC front-panel I/O cutouts to enable attachment of up to two PMC expansion cards. When installed, these cards access a PCI bus through compatible connectors provided on the host board. See Section 3.4.4, *PMC and PIM Interface*.

### 3.5.3 Slot IO

The CP2080 board is fitted with an 80-pin connector in the J3204 location. This connector is an IO connector used by Sun Microsystems™ only.

### 3.5.4 Backplane IO

Most of the IO channels to or from the board are passed to CompactPCI connectors J3 and J5; these channels are accessible from external connections on a transition card connected at the rear of the CompactPCI backplane. Connector J4 is not populated on these host boards to prevent contention with H110-compliant backplane signals. Contact assignments for these connectors are shown in Chapter 8, *Connectors, Pinouts and Switch Settings*. For location of the connectors, see *Figure 3-2* and *Figure 3-3*.

#### 3.5.4.1 J3 Signals

The user-defined PMC I/O signals from the two PMC Jn3 connectors pass to their external interface through the J3 CompactPCI backplane connector (see Chapter 8, *Connectors, Pinouts and Switch Settings*).

#### 3.5.4.2 J5 Signals

The following signal sets pass through the J5 CompactPCI backplane connector to connect to an external interface connector on the transition card:

- Two TP Ethernet channels from the PHYs
- Two serial channels
- Two USB channels
- An $I^2C$ channel
- PMC I/O

## 3.6 System Management Controller

The System Management Controller (SMC) subsystem is one of the most important components of the system board. This subsystem provides a variety of service functions related to assuring availability of the system. These functions contrast with the board functions that execute applications.

The SMC subcircuit consists of a small microcontroller with an SRAM for a software stack and non-volatile memory for program storage and data logging. The SMC is modular in character but it is physically embedded into the circuitry of the Netra CP2060 / CP2080 board. *Figure 3-21* shows its functional relationship with the system.

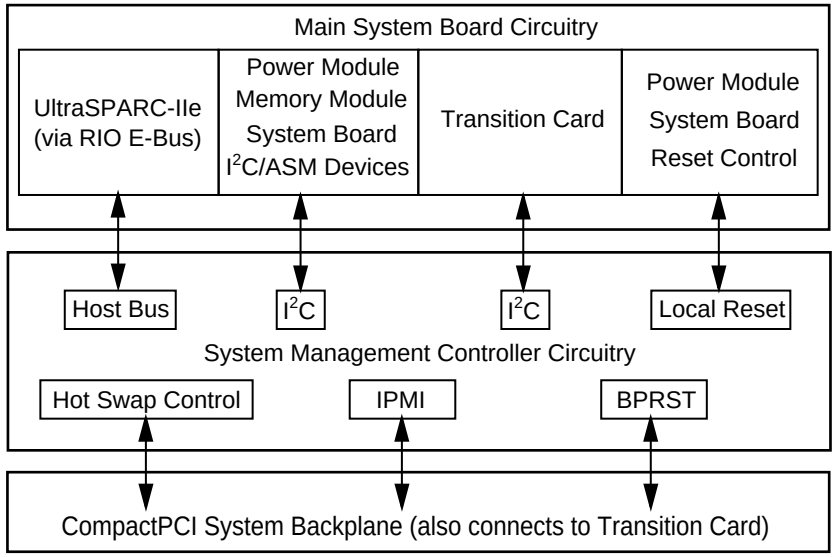



**Figure 3-21** System Management Controller Interface

The SMC hardware and firmware implements the functions of System Management and Hot Swap control.

---

**Note –** Although the hardware and firmware functions are architecturally separate, reference to the SMC subsystem in this document—whose description is hardware oriented—refers to both functions.

---

The SMC controls the on-board CompactPCI interface components for the Hot Swap process. It coordinates the state of the 21554 PCI bridge, the arbiter functions, and the switched connection of critical CompactPCI signals to the bus.

In performing these functions, the design maintains conformance with the PICMG CompactPCI core specification and the PICMG CompactPCI Hot Swap specification; see *Bibliography* for references to these documents. The main features that are supported by the SMC subsystem are:

- Coordinates and controls local resets on the system board during POR, watchdog timeout, software initiated reset, and as a result of user intervention such as pushbutton reset or backplane reset.

- Operation as either a system host board or as a satellite board in standard CompactPCI backplanes.
- Supports a command and communications interface between the UltraSPARC-IIe host processor and the SMC microcontroller by means of an interconnecting E-Bus. This interface acommodates a command suite from the UltraSPARC processor to the SMC and supports bi-directional interrupt between the UltraSPARC processor and the SMC.
- Supports signals that are critical to system configuration or Hot Swap reconfiguration. These signals accomplish CompactPCI reset control, bus enumeration, and configuration of CompactPCI bus arbitration. These signals apply differently depending upon whether the board acts as a system host board or a satellite board.
  - a satellite board responds to the CompactPCI RST , ENUM, HEALTHY and BD_SEL signals from the active system host board (provided that this host provides HA Hot Swap functions through the CompactPCI bus). See also Section 6.6.2, *ENUM Monitoring*).

---

**Note –** These boards do not provide HA Hot Swap control for peripherals. It also sets a REQ signal to the arbiter on the system host board and awaits a corresponding GNT from it.

---

  - a system board provides the CompactPCI peripheral reset signal PCI_RST and reads the CompactPCI bus enumeration signal ENUM. Its CompactPCI arbiter is configured to issue GNT signals to peripherals in response to their REQ requests.

  The role that the Netra board assumes (system host board or satellite board) depends upon the slot into which it is installed.
- Implements a two-level watchdog timer for the SMC processor and for the host processor
- Supports two $I^2C$ ports. One of these carries an IPMI bus that is routed through CompactPCI backplane connector J1 to enable communication with other SMCs in the system. The other $I^2C$ port serves a multiplexer which splits its input into two channels, one of which provides communication with on-board devices, for example, for temperature or FRU information; the other channel, for user functions, is passed out of J5 connector.
- An EEPROM for storing non-volatile data such as the host board ID MAC address.
- Supports two system controller models of CompactPCI Hot Swap system architecture —see Section 3.9, *Hot Swap*; also see *Bibliography* for a reference to the PICMG CompactPCI Hot Swap specification.
- Communicates with other SMCs in the cPCI system using the IPMI protocol over a backplane link, in accordance with the PICMG CompactPCI Hot Swap specification. It:

- Assumes a Baseboard Management Controller (BMC) role when the host board is a System Slot Controller, that is, installed in a system slot
- Responds to commands from the active BMC when the host board is installed in a peripheral slot

- Local Advanced System Monitoring (ASM); ASM is a management scheme that monitors—and controls where appropriate—local board or chassis "housekeeping" functions through the on-board $I^2C$ interface. Such functions include (see also Section 6.6, *ASM Support at OBP*):
  - temperature sensing
  - power supply voltage sensing
  - power supply module on / off control
  - memory module and board ID detection
- Flash update: The SMC firmware supports external update of its flash RAM (see Chapter 6, *Firmware*).
- Peripheral Management interface; this includes the functions:
  - IPMI communications with Baseboard Management Controller
  - Handling of Hot Swap (HEALTHY / ENUM / PRESENT) related signals
  - Receiving system and CompactPCI reset events to generate local board reset

For full details on SMC and reset information, please refer to Chapter 6, *Firmware* and the Netra CP2060 / CP2080 website:

*http://www.sun.com/microelectronics/commprovider/cp2060/*

### 3.6.1 Watchdog Timer

In the Netra CP2060 / CP2080 board, the SMC implements a two-level watchdog timer. The host-SMC command interface, defines communication between host and SMC. The host and the SMC constantly communicate with each other when the watchdog timer is enabled. The SMC monitors the heartbeat of the CPU processor host. The heartbeat is sent in the form of a reset watchdog timer that is sent from the CPU to the SMC. It must be programmed to ensure that it does not get too close to the expiration. There should be some time accounted for the latency overhead or any unexpected event that may delay transmission of the heartbeat. For full details on programming the watchdog timer, please refer to the Netra CP2060 / CP2080 website:

*http://www.sun.com/microelectronics/commprovider/cp2060/*

The two levels of the watchdog timer are as follows:

- the countdown register timer (16 bits, 100 msec. resolution)
- the pre-timeout timer (1 sec. resolution)

The two watchdog timers are enabled by messages sent over the host-SMC command interface using the set watchdog timer command. The commands enabled in the host-SMC command interface for watchdog timer functionality are:

- Reset watchdog timer
- Set watchdog timer
- Get watchdog timer

The uses of these functions are shown in *Table 3-1*.

**Table 3-1** Host-SMC Commands for Watchdog Timer

| Host-SMC Command | Uses |
|---|---|
| Reset watchdog timer | Starting-restarting watchdog timer from the initial countdown value |
| Set watchdog timer | Initializing, configuring and stopping the watchdog timer |
| Get watchdog timer | Retrieval of current settings and present timer value of watchdog timer |

## 3.7 Resets and Interrupts

This section provides details on resets and interrupts.

## 3.7.1 Resets

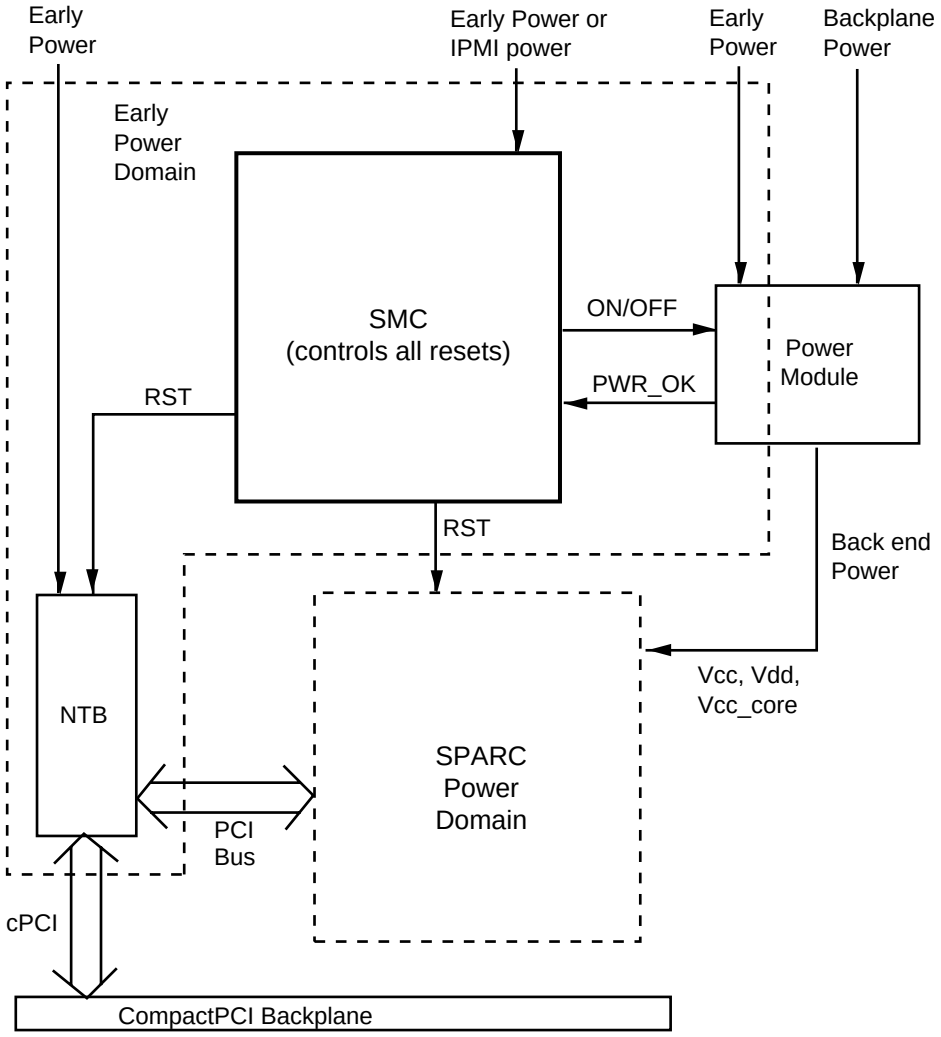



**Figure 3-22** Simplified Reset Paths

Parts of the system are powered by *early power* before the SPARC domain receives power (backend power). See Section 3.8, *Power Subsystem*. At the onset of early power, the SMC is reset by its component microcontroller. When backend power rails are at their specified voltages and if the SMC has tonrn on the power module turned on, the SMC receives the PWR_OK signal and in turn resets the backend members of the system. Note that:

- When the SMC is reset, the whole system is reset
- When the CPU is reset, the CPU IO and the 21554 NTB is reset
- The SMC can reset the 21554 NTB without resetting the CPU

For detailed information on configurable reset implementation by SMC firmware, please see Chapter 6, *Firmware*.

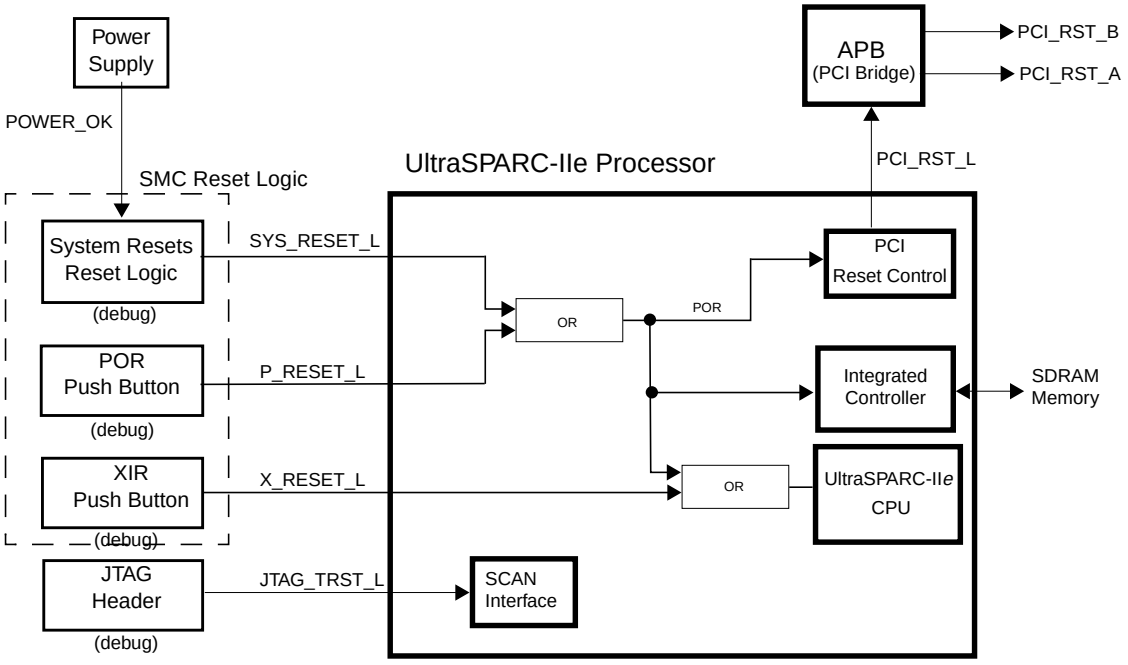



**Figure 3-23** Simplified CPU Subsystem Reset Architecture

## 3.7.2 Interrupts

Interrupts to the UltraSPARC-IIe processor are listed in Chapter 4, *Interrupts and Addresses*. These are processed and encoded by the I-Chip2 ASIC. This device assigns equal priority to all interrupting devices. When two devices need servicing at the same time, the I-Chip prioritizes using its internal round-robin scheduling scheme. The resultant vector is passed to the processor as a 6-bit parallel word. The ultimate interrupt priority is resolved in the UltraSPARC-IIe processor.

# 3.8 Power Subsystem

*Figure 3-24* shows a simplified schematic diagram of the power subsystem. This subsystem provides for powering the board in a way that supports a Hot Swap environment.

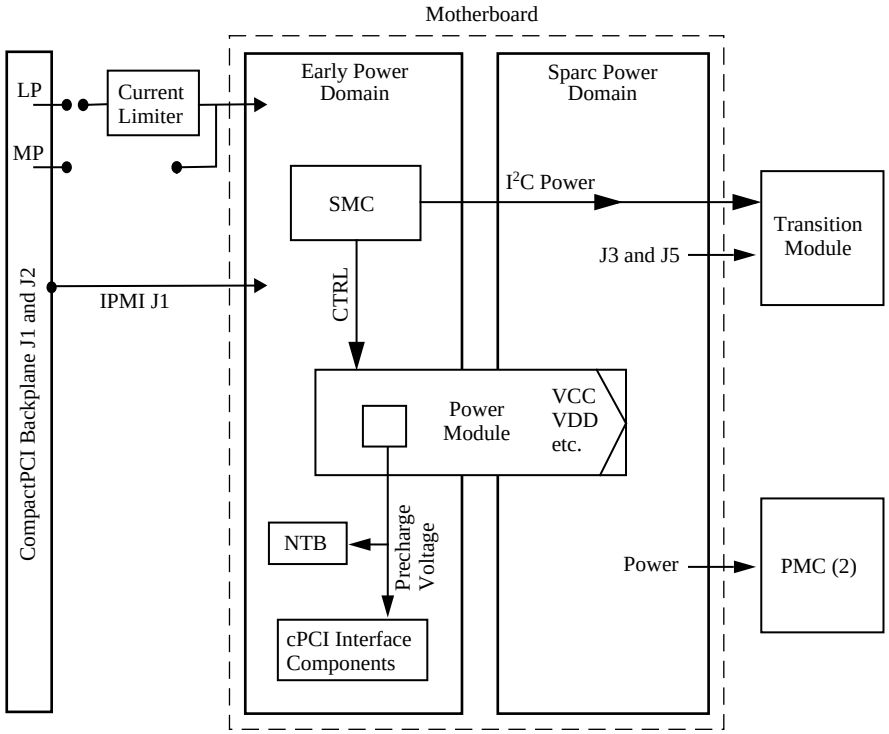



**Figure 3-24** Power Distribution Block Diagram

**Note –** In *Figure 3-24* I2C power is derived from Early Power.

The Netra CP2060/CP2080 board sequences power in two time-separated domains:

- Early Power domain
- SPARC Power domain—this is the *Backend Power* in the PICMG Hot Swap specification.

Early power is applied to the board from backplane long pins (LP in the figure) as the board is inserted. Early power current flows to board subsystems:

- Power Module—supplies precharge current to the CompactPCI bus interface components
- SMC—needed to control logical state of the CompactPCI interface circuits as they are connected
- IPMI/$I^2C$ subsystems—needed for management/monitoring functions at this stage; $I^2C$ power also extends to the transition card
- The 21554 NTB and CompactPCI Interface components—must be placed in a known state during attachment to the CompactPCI bus

## 3.8.1 Power Module

*Figure 3-25* shows a schematic diagram of the power module. This subassembly is integral with the Netra CP2060/CP2080 board.

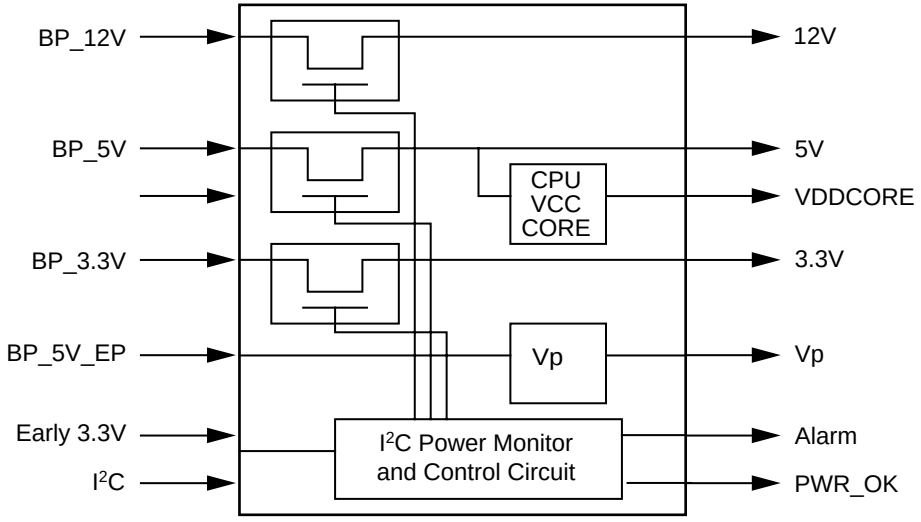



**Figure 3-25** Power Module Interface

This subsystem performs the following functions:

- Generates Vp, the CompactPCI Hot Swap precharge bias voltage using early power

- Generates $V_{DDCORE}$ , the UltraSPARC processor core voltage supply.
- Controls and gates 5V, 12V, 3.3V and -12V
- Automatically shuts down in case of overcurrent or overvoltage
- Asserts the PWR_MOD_OK signal

The power module is controlled by the SMC and the power ON/OFF signal. Functions controlled include core voltage, output level, and module on or off state. There are also automatic controls within the power module, for example, overcurrent shutdown and voltage regulation.

The power module has a DIP switch with six preset default settings. These switches are for factory use only (see *Figure 3-26* for location). The user must not change DIP switch settings.

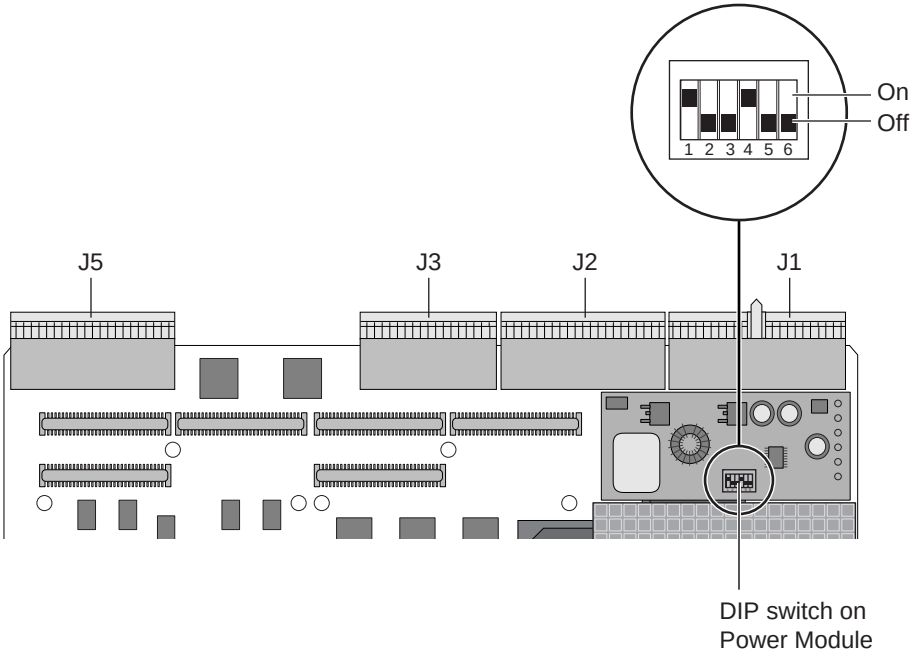



**Figure 3-26** DIP switch settings on Power Module

## 3.8.2 Early Power and IPMI Power

In the event of a failure of system power for the backplane, the SMC can use IPMI power, typically supplied from an uninterruptible power supply (UPS), instead of Early Power from the CompactPCI backplane. The backplane is provided with IPMI power pins for this purpose. *Figure 3-27* shows the circuit arrangement that selects between these power sources.

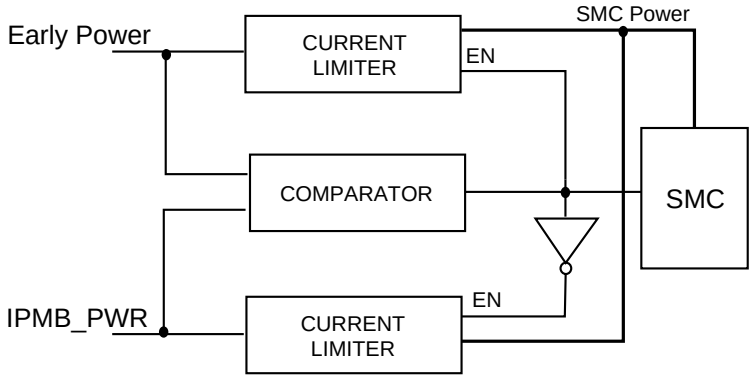



**Figure 3-27** Selection between Early Power and IPMI Power

## 3.8.3 Transition Card Power Distribution

*Figure 3-28* shows the power rail routing to the transition card. The XCP2060-TRN I/O Transition Card is powered from the Netra CP2060/CP2080 board rather than directly from the backplane. The transition card must always be connected to the backplane before the chassis is powered. Always install the transition card before the Netra CP2060/CP2080 board in the chassis. For details on using a transition card, please see Chapter 5, *Installation*.

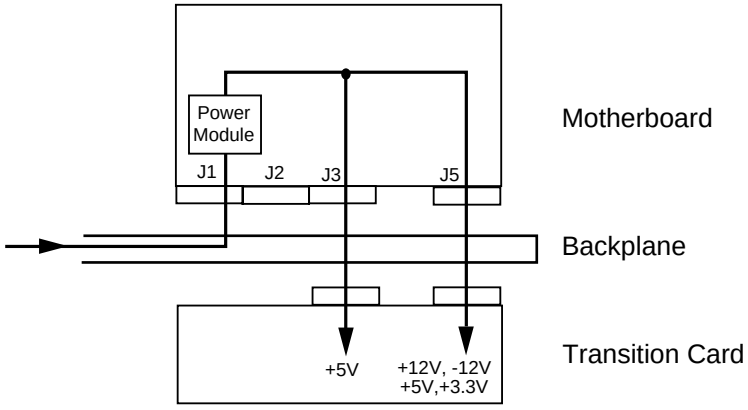



**Figure 3-28** Transition Card Power Supply Routing

---

**Note –** Some V(I/O) power lines are routed into J2 of the motherboard; this is not shown in the figure for clarity.

---

# 3.9 Hot Swap

This section briefly discusses the hot swap support on the Netra CP2060/CP2080 boards.

## 3.9.1 CP2060/CP2080 Hot Swap Support

*Table 3-2* lists the hot swap support details when a Netra CP2060/CP2080 board functions as a system host board or as a satellite board.

**Table 3-2** Netra CP2060/CP2080 Board Hot Swap Support

| Netra CP2060/CP2080 Board Role | Basic Hot Swap | Full Hot Swap | HA Hot Swap |
|---|---|---|---|
| System Host Board Role | yes | yes | no |
| Satellite Board Role | yes | yes | no |

## 3.9.2 Hot Swap Architecture and System Models

The following section provides hot swap information and a brief description of the different hot swap models.

See *Bibliography* for a reference to the PICMG CompactPCI Hot Swap Specification which provides a detailed description of this subject. In general, the Hot Swap process includes the orderly connection of the hardware and software. This process uses:

- *hardware connection control*—to connect the hardware in an orderly sequence; this process includes the use of backplane pins of different lengths to accomplish signal sequencing to protect the hardware and avoid corrupting the backplane bus.
- *software connection control*—to bind software device drivers to the peripheral hardware (peripheral is used here to mean peripheral with respect to the Hot Swap controller—in the case of Sun CompactPCI host boards this also means satellite board to the system host board because the system host and Hot Swap controller functions lie on the same board.) once it has been connected, or unbind the drivers from the hardware before disconnection. A failure to achieve this control properly results in a non-functional peripheral or a system panic.

There are three models of Hot Swap described in the PICMG CompactPCI Hot Swap Specification: Basic Hot Swap, Full Hot Swap, and HA Hot Swap.

CHAPTER 4

# Interrupts and Addresses

This chapter provides information on the following topics:

- CompactPCI interface
- Registers
- Interrupts

## 4.1 CompactPCI Interface

This section provides information on CompactPCI and the Netra CP2060/CP2080 board interface requirements specifications and CompactPCI signal interface.

### 4.1.1 CompactPCI Interface Requirements

These are the requirements for Netra CP2060/CP2080 boards as defined by the PICMG 3.0 CompactPCI and Netra CP2060/CP2080 design requirements specifications:

**Table 4-1** Compact PCI Interface Requirements

| Requirement | Description |
|---|---|
| Bus termination | 10 ohm cPCI series termination resistor shall be located 0.600 max. from J1/J2 pin on all required signals. |
| Stub Length | cPCI pull-up stub length 0.500 inches max. |
| 5V VIO | Provides 1.0K ohm +/-1% pull-up for all required CompactPCI bus signals for use in 5V CompactPCI signaling environment. |

**Table 4-1** Compact PCI Interface Requirements

| Requirement | Description |
|---|---|
| System and Peripheral Slot operation | Provide control to disable both pull-ups (In a CompactPCI system, only the board in the system slot can provide the bus pull-ups). |
| Hot Swap | Provide 1V +/-20% precharge bias voltage (Vp) for all required CompactPCI bus signals. |
| Max capacitive load per pin and System and Satellite Slot operation | Provide an auxilliary output for selected CompactPCI bus signals (those that are shared between the bridge and external arbiter) |

## 4.1.2 CompactPCI Signal Interface

The tables in this section list the cPCI signal interface description and the cPCI connector power signal interface. The primary side of the bridge is attached to the 64 bit CompactPCI bus.

**TABLE 4-2** CompactPCI Interface

| Signal | Description | Type | Notes |
|---|---|---|---|
| P_AD<63..0> | Addr/Data Bus | I/O | |
| P_CBE<7..0> | Command/ Byte Enable# | I/O | |
| P_CLK | Clock Input | I | 33 MHz Compact PCI Bus Clock |
| P_IDSEL | ID Select | I | |
| P_INTA_L | Primary Bus Interrupt | OD | Needs external pullup. Assert when:<br>-Primary doorbell register bit set<br>-I20 outbound queue not empty<br>-Subsystem event bit set |
| P_GNT_L | Bus Grant | I/O | Shared with arbiter |
| P_REQ_L | Bus Request | I/O | Shared with arbiter |
| P_PAR | Parity for lower 32 bits | I/O | |
| P_PAR64 | Parity for upper 32 bits | I/O | Requires external pullup. |
| P_PERR_L | Parity Error | I/O | Requires external pullup. |
| P_SERR_L | System Error | I/O | Requires external pullup. |
| P_RST_L | Bus Reset | I | |

**TABLE 4-2** CompactPCI Interface

| Signal | Description | Type | Notes |
|---|---|---|---|
| P_VIO | Signaling Environment 3.3V or 5V | I | |
| P_DEVSEL_L | Device Select | I | Requires external pullup. |
| P_FRAME_L | Frame | I | Requires external pullup. Shared with arbiter. |
| P_STOP_L | Stop | I | Requires external pullup. |
| P_IRDY_L | Initiator Ready | I/O | Requires external pullup. Shared with arbiter. |
| P_TRDY_L | Target Ready | I/O | Requires external pullup. Shared with arbiter. |
| P_REQ64_L | 64 bit transfer request | I/O | Requires external pullup. |
| P_ACK64_L | 64 bit transfer ack | I/O | Requires external pullup. |

**Table 4-3** cPCI Connector Power Signal Interface

| Voltage | cPCI Pin(s) | Net Name | Notes |
|---|---|---|---|
| 3.3V | J1-C6 & C22 | BP_EP_3.3V | Long pin |
| 3.3V | J1-A15, A17, A19, A21, A23, C10, C18, & D25 | EP_3.3V | Medium pin |
| 5V | J1-D3 & D23 | BP_EP_5V | Long pin |
| 5V | J1-A1, A25, B2, B24, E1, & E25 | EP_5V | Medium pin |
| +12V | J1-D1 | BP_12V_POS | Medium pin |
| -12V | J1-B1 | BP_12V_NEG | Medium pin |
| 3.3V or 5V | J1-C4 | BP_EP_VIO | Long pin |
| 3.3V or 5V | J1-C8, J1-C16, J1-C24, J2-A4, J2-C5, J2-C7, J2-C9, J2-C11, J2-C13 | EP_VIO | Medium pin |

**Note –** The early power voltages supply critical circuits such as SMC, cPCI interface circuits, power module control circuit.

# 4.2 Interrupts

The Netra CP2060/CP2080 board interrupts are listed in *Table 4-4*. These are processed and encoded by the I-Chip2 ASIC. See Chapter 3, *Hardware and Functional Description* for more information on Interrupts.

**Table 4-4** Interrupt Assignments

| Offset | Interrupt | INT# | Priority |
|---|---|---|---|
| 0 | CPCI_INTA[1] | 7 | 7 |
| 1 | CPCI_INTB | 5 | 5 |
| 2 | CPCI_INTC | 15 | 5 |
| 3 | CPCI_INTD | 2 | 2 |
| 4 | SPARC_H_INT[2] | F | 7 |
| 5 | | D | 5 |
| 6 | | LD | 5 |
| 7 | | A | 2 |
| 8 | PCIO-2_A_ENET | 17 | 6 |
| 9 | | 38 | 5 |
| A | PMC1_INT_A | 10 | 2 |
| B | PMC1_INT_B | 12 | 1 |
| C | PMC1_INT_C | 18 | 6 |
| D | Not on IChip2 | 39 | 4 |
| E | | 0 | 2 |
| F | ENUM_L[3] | 1A | 1 |
| 10 | DUART_SER_A | 6 | 6 |
| 11 | MCA_INT_L | 4 | 4 |
| 12 | | 3 | 3 |
| 13 | | 1 | 1 |
| 14 | DUART_SER_B | E | 6 |
| 15 | PMC1_INT_D | C | 4 |
| 16 | PMC2_INT_A | B | 3 |
| 17 | | 9 | 1 |
| 18 | | 16 | 6 |

**Table 4-4** Interrupt Assignments

| Offset | Interrupt | INT# | Priority |
|---|---|---|---|
| 19 | PCIO-2_B_EBUS | 14 | 4 |
| 1A | | 13 | 3 |
| 1B | | 11 | 1 |
| 1C | PCIO-2_B_ENET | 1E | 6 |
| 1D | PCIO-2_A_EBUS | 1C | 4 |
| 1E | SPARC_L_INT[2] | 1B | 3 |
| 1F | | 19 | 1 |
| 20 | PMC2_INT_B | 20 | 3 |
| 21 | | 21 | 3 |
| 22 | PMC2_INT_C | 22 | 2 |
| 23 | | 24 | 8 |
| 24 | PCIO-2_A_USB | 1F | 7 |
| 25 | | 25 | 8 |
| 26 | PCIO-2_B_USB | 28 | 7 |
| 27 | PMC2_INT_D | 29 | 8 |
| 28 | I2C_GLOBAL_INT | 2A | 2 |
| 29 | | 2B | 4 |
| 2A | | 2C | 4 |
| 2B | SYNC_SER_L1 | 2D | 7 |
| 2C | | | RES |
| 2D | | | RES |
| 2E | | | RES |
| 2F | | | RES |
| graphic 1<br>23 from INR | CPCI_SERR_L[4] | 23 | 5 |
| graphic 2<br>26 from INR | | 26 | 5 |

1. CPCI_INT_A is shared with the Bridge Secondary side interrupt.
2. SPARC_L_INT and SPARC_H_INT are driven by the SMC module.
3. ENUM_L is masked at the PLD.
4. CPCI_SERR_L is masked at the PLD.

# 4.3 Chip-Select PLD Registers

The *Table 4-5* lists the chip-select PLD registers.

**Table 4-5** Chip-Select PLD Registers

| EBus Address | R/W | Name | Description |
|---|---|---|---|
| 0x20.000 | W | DUART_RESET | Any write operation to this register toggles the DUART reset line. |
| 0x20.0001 | R/W | PMC_BUSMODE | Bit 4 :<br>PMC_BUSMODE4_L. R/W; boot default = 0.<br>Bit 3 :<br>PMC_BUSMODE3_L. R/W; boot default = 0.<br>Bit 2 :<br>PMC_BUSMODE2_L. R/W; boot default = 1.<br>Bit 1 :<br>PMC_BUSMODE1_L. Read only; writes are ignored.<br>Bit 0 :<br>PMC_BUSMODE0_L. Read only; writes are ignored. |
| 0x20.0002 | R/W | WRITE_PROTECT | Bit 1 : $I^2C$ write protect.<br>1 = protected.<br>0 = unprotected.<br>Boot default is 0. |
| 0x20.0003 | R | PLD_REV | Read returns the PLD firmware revision, e.g. 0xB3. |
| 0x20.0004 | R/W | SPARC_INT | SMC_SPARC_L_INT_L mask bit.<br>SMC_SPARC_L_INT_L enters the PLD.<br>PLD_SMC_SPARC_L_INT_L is the masked output.<br>0 = masked HIGH.<br>1 = not masked.<br>Boot default is 1. |

**Table 4-5** Chip-Select PLD Registers

| EBus Address | R/W | Name | Description |
|---|---|---|---|
| 0x20.0005 | R/W | INTx | cPCI interrupts mask register.<br>cPCI interrupts enter PLD as SW_CP_INTx_L.<br>Masked outputs are PLD_CP_INTx_L.<br>Mask behavior:<br>0 = masked HIGH.<br>1 & CP2060/CP2080 is SBC = unmasked.<br>1 & CP2060/CP2080 is not SBC = TRISTATE.<br>Bit assignments:<br>Bit 3 = INTA_L.<br>Bit 2 = INTB_L.<br>Bit 1 = INTC_L.<br>Bit 0 = INTD_L. |
| 0x20.0006 | R/W | ENUM | cPCI ENUM mask bit.<br>ENUM_L is the input to the PLD.<br>PLD_ENUM_L is the masked output.<br>0 = masked HIGH.<br>1 = unmasked.<br>Boot default = 0. |
| 0x20.0007 | R/W | TEST | For development use. May be written with any value and read. |
| 0x20.0008 | R/W | GREEN_LED | Bits 1,0:<br>00 = LED pulses 'heartbeat'<br>01 = LED on<br>10 = LED flashes, 2Hz, 50% D.C.<br>11 = LED off. |
| 0x20.0009 | R/W | USERFLASH_SELECT | Bit 0 :<br>0 = Sharp user flash (default). R/W<br>1 = Intel user flash (any size). R/W<br>Bit 7 :<br>PLD_FLASH1_SEL (Read only).<br>Bit 6 :<br>PLD_FLASH0_SEL (Read only). |

**Note –** The registers in the chip-select PLD are mirrored in the EBus address space.

# 4.4 SMC PLD Registers

**TABLE 4-6** SMC PLD Registers

| EBus Address | R/W | Name | Description |
|---|---|---|---|
| 0x00.000 | | SMC_INT0 Trigger Register No. 1 | Bit 7 : PWR_MOD_OK signal.<br>Bit 6 : Latched cPCI RST#<br>0 = cPCI RST# was asserted.<br>1 = cPCI RST# has not been asserted.<br>Bit 5 : PB_RST_L signal.<br>Bit 4 : PB_ABORT_L signal.<br>Bit 3 : SPARC_RST_21554_L signal.<br>Bit 2 : BKRST_IN_L signal.<br>Bit 1 : BP_DEG_L signal.<br>Bit 0 : BP_FAL_L signal.<br><br>Any of the bits asserting LOW initiates an assertion of SMC_INT0_L. |
| 0x00.0001 | R | SMC_INT0_L Trigger Register No. 2 | Bit 7 : SERR_L signal INVERTED.<br>Bit 6 : Non-latched PCI RST# signal.<br>Bit 5 : IPMI power status<br>0 = no ipmi_pwr in<br>1 = ipmi_pwr in<br>Bit 4 : Option switch #2 status.<br>Bit 3 : Option switch #1 status.<br>Bit 2 : Reserved.<br>Bit 1 : Reserved.<br>Bit 0 : Reserved. |
| 0x00.0002 | R / W | General purpose register | Bit 7 : bp_GPIO_j4e1 GPIO.<br>Bit 6 : bp_pwroff_j4c10 power.<br>Bit 5 : sys_rst_set.<br>Bit 4 : bkrstout_set.<br>Bit 3 : 21554_rst_set.<br>Bit 2 : xir_out_set.<br>Bit 1 : pb_rst_set.<br>Bit 0 : rst_j1c5_set.<br><br>All bits default HIGH. |
| 0x00.0003 | R / W<br>R | Address register | Bits 7-5 : Address FIFO.<br>Bits 4-0 : ADDR3 Register.<br>Bit 4 : block PRST, default LOW.<br>Bit 3 : block cPCI RST#, default HIGH.<br>Bit 2 : enable NVRAM, default LOW.<br>Bit 1 : PWR_MOD_ON, default HIGH.<br>Bit 0 : SYSEN_ON_L, default LOW. |

CHAPTER 5

# Installation

This chapter describes some of the typical system configurations in which the Netra CP2060/CP2080 boards may be used. It describes available system configurations, a summary of installation procedure, Netra CP2060/CP2080 board configuration, XCP2060-TRN I/O transition card configuration and how to install the serial EEPROM.

## 5.1 System Configurations

The Netra CP2060/CP2080 boards can be mounted in various enclosures, such as shown in *Figure 5-1*. They can be deployed in various electrical configurations to suit each end-user requirement. For example, the host board can be used with a transition card and configured to boot from a network as a diskless client. Alternatively, industry-standard PMC and PIM hardware from Independent Hardware Vendors (IHV) can be employed to provide local disk IO which may optionally be used as a boot path. The installation procedure is essentially independent of the type of enclosure, whether a floor-mounting rack or a bench-top cabinet is employed. Memory is only user configurable in the Netra CP2080 board; the Netra CP2060 board has fixed on-board memory.

### 5.1.1 Order Items

Sun provides these items to customer order:

- Host board: SEUCP2060-500: Netra CP2060 (with on-board, soldered memory)
- Host board: SEUCP2080-500: Netra CP2080—supplied with no on-board memory
- Netra CP2080 memory options (see *Table 2-5*):

- XCP2000-MEM-256MB: 256MB memory mezzanine card
- XCP2000-MEM-512MB: 512MB memory mezzanine card
- XCP2000-MEM-1GB: 1GB memory mezzanine card

- XCP2060-TRN IO Transition Card—shown in *Figure 5-2*; this assembly is compatible with the Netra CP2060 or CP2080 host boards (see the *Bibliography* for references to the XCP2060-TRN I/O transition card manual).

A compatible transition card is required to be used with the Netra CP2060 and Netra CP2080 boards for I/O access. The transition card enables access to the network, to a boot device and to a console terminal.

The customer must procure the following components as required:

- Solaris 8 operating environment, Release 1/01 support (minimum)
- Serial terminal or terminal emulation for console output
- Cables for terminal and network connection
- PIM and PMC hardware

**Table 5-1** CompactPCI System and Other Minimum Requirements Dependant on Board Function

| Requirements | Netra CP2060/CP2080 as System Host Board | Netra CP2060/CP2080 as Satellite Board |
|---|---|---|
| CompactPCI system box for 6U boards (includes chassis, backplane, power supply[1])[2] | yes | yes |
| Console output device/serial terminal | yes | yes |
| Boot device (such as hard drive) | yes | yes |
| Peripheral device for network access | yes | yes |
| System controller | no | yes |

1. See Chapter 2, *Specifications* to ensure that your system enclosure meets the power supply and cooling requirement specifications.
2. See *Figure 5-3* and *Figure 5-4* for a typical arrangement

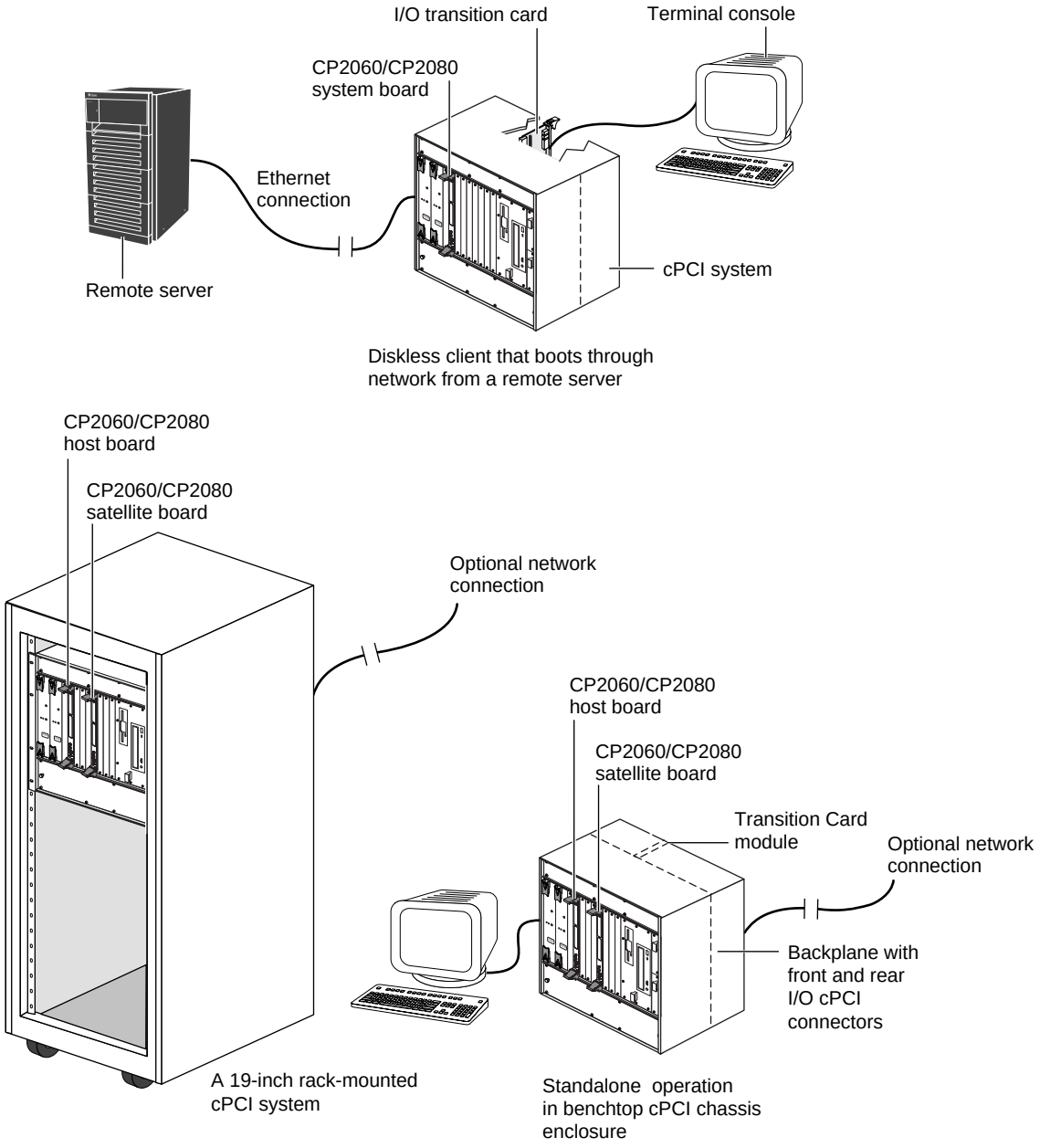



**Figure 5-1** Examples of Netra CP2060/CP2080 Mounting Configurations

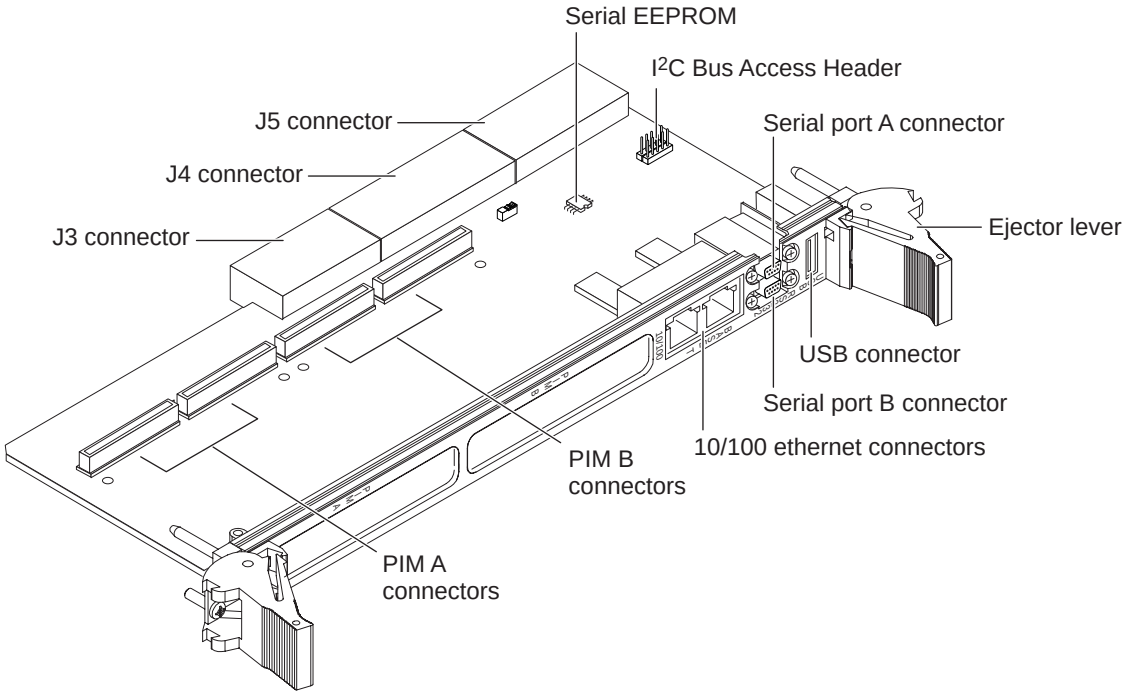



**Figure 5-2** XCP2060-TRN Transition Card with Location of On-Board Components

The Netra CP2060 and CP2080 boards require the addition of Independent Hardware Vendor (IHV)-built PMC modules to access IO on their front panels. PMC modules decode their custom IO from the Netra board's on-board PCI bus B signals; see Section 3.4.4, *PMC and PIM Interface*).

The optional XCP2060-TRN Transition Card from Sun Microsystems installs from the rear of the CompactPCI enclosure, opposite the host board whose IO it captures. The transition card connects with the host CompactPCI P3 and P5 connectors through the backplane pins and carries two serial ports and a USB port out to its rear-panel flange. *Figure 5-3* and *Figure 5-4* shows the physical relationship between host boards, transition cards, and the backplane in a typical system.

---

**Note –** To meet emission standards, the user needs to use shielded cables for serial, Ethernet and USB ports on the transition card and the shield should be grounded at both ends.

---

The transition card can also be fitted with IHV PCI Interface Modules (PIMs) which are configured to bring IO channels to the unit rear panel. A unit of PIM hardware is a kit that includes a card for the PMC slot and a card for the PIM slot on the transition card. A PIM is essentially a rear-panel extension added to a PMC module. Even if the PMC card (used with the PIM) has flange (front IO) connectors, configuring PIM (rear) connector IO excludes front PMC output.

The customer can order the XCP2060-TRN Transition Card, build a custom card, or buy from an Independent Hardware Vendor (IHV). A minimal set of IO must provide for a boot path for the host board and for a path for console IO to deliver commands and to read board and system status.

Possible boot and console configurations are described in *Table 5-2*. Sun Microsystems provides the host boards and a compatible XCP2060-TRN Transition Card (see *Figure 5-2* and the XCP2060-TRN I/O Transition Card Manual referenced in *Bibliography*) for Netra CP2060/CP2080 boards. This transition card brings out 10/100 Ethernet RJ45 ports from the host which can be used to accomplish network boot as a diskless client. The other configurations require IHV hardware.

---

**Note –** A PIM kit is IHV supplied as a PMC module without front panel IO and a matching PIM module—also see *Table 2-6* for PMC capacity of host boards.

---

**Table 5-2** Netra CP2060 and Netra CP2080 IO Configurations

| IO | Hardware Required | Description |
|---|---|---|
| Ethernet | XCP2060-TRN IO Transition Card—supplied as an option | default boot path uses Ethernet port on Transition Card; host runs in diskless client configuration |
| SCSI | XCP2060-TRN IO Transition Card; PMC SCSI IO | may be used for local boot; requires optional transition card with PMC SCSI IO |
| Serial data | XCP2060-TRN IO Transition Card | serial port A on optional transition card is path of default console IO (see *Figure 5-2* for location) |
| SVGA video | XCP2060-TRN IO Transition Card and PMC video graphics controller, or cPCI video controller card | cPCI card video controller takes one cPCI slot and attaches directly to cPCI backplane |
| USB | XCP2060-TRN IO Transition Card | can be used for keyboard IO for use with video graphics |

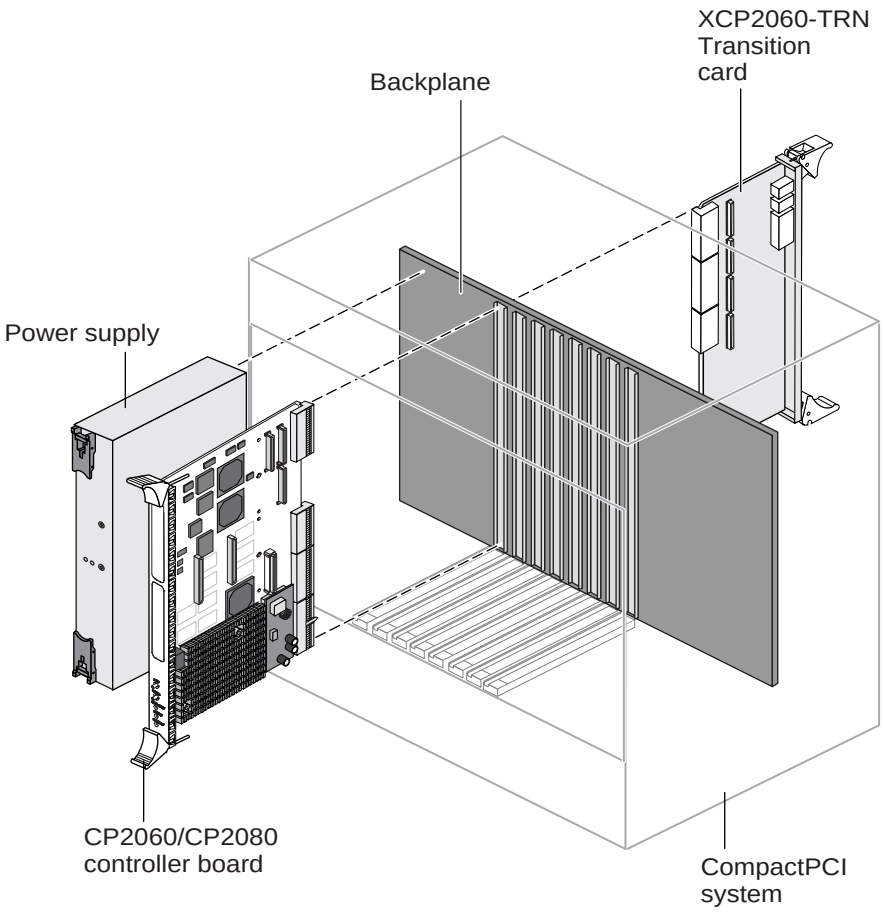



**Figure 5-3** Typical cPCI System Illustrating the Netra CP2060/CP2080 Board in System Board Role with XCP2060-TRN Transition Card

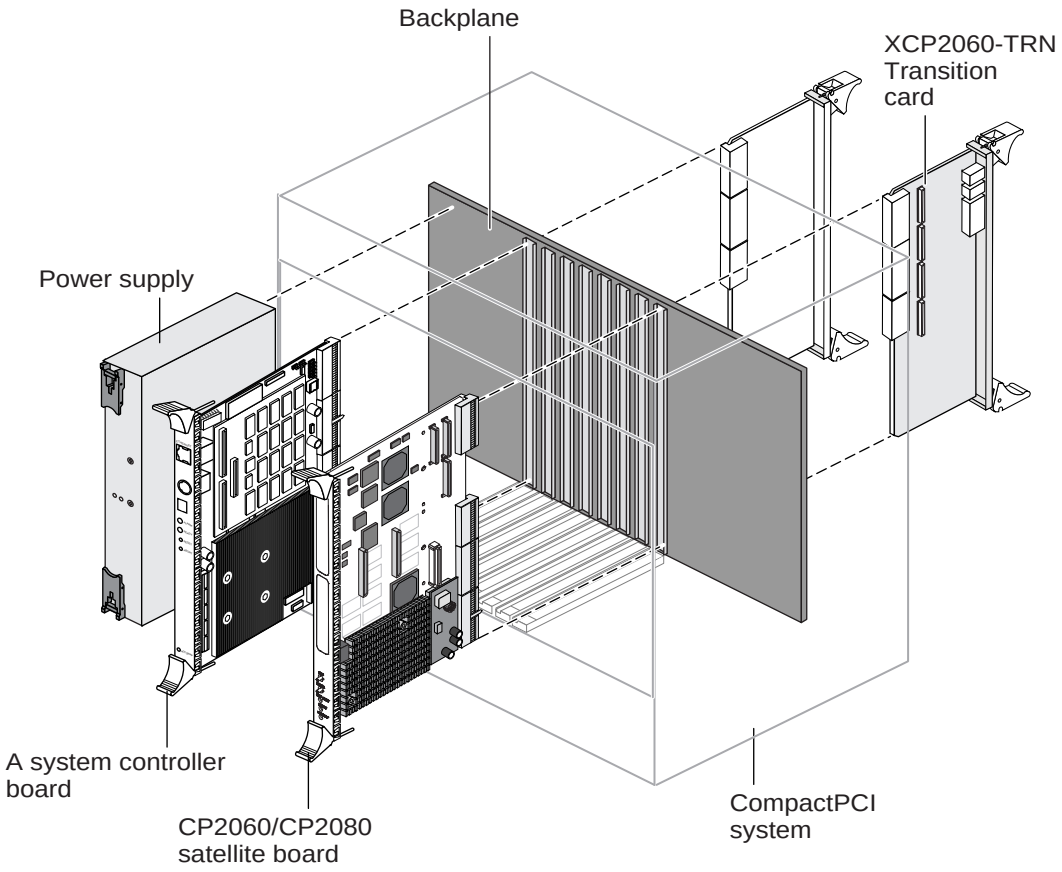



**Figure 5-4** Typical cPCI System Illustrating the Netra CP2060/CP2080 Board in Satellite Board Role with XCP2060-TRN Transition Card

# 5.2 Equipment and Operator Safety

Read these safety statements carefully before you install or remove any part of the system.

Depending on the particular chassis design, operations with open equipment enclosures can expose the installer to hazardous voltages with a consequent danger of electric shock. Ensure that line power to the equipment is disconnected during operations that make high voltage conductors accessible.

The installer should be familiar with commonly-accepted procedures for integrating electronic systems and should also be acquainted with the general practice of Sun systems integration and administration. Although parts of these systems are designed for hot swap operation, other conponents must not be subjected to such stresses. Work with power connected to a chassis only when necessary and follow these installation procedures to avoid equipment damage.

This equipment is sensitive to damage from electrostatic discharge (ESD) from clothing and other materials. Use the following ESD preventive measures during an installation.

- If possible, disconnect line power from the equipment chassis when servicing a system or installing a hardware upgrade. If the chassis cannot be placed upon a grounded ESD protective mat, connect a grounding strap between the facility electrical input ground (usually connected to the equipment chassis) and facility electrical service ground.
- Use an ESD-protective wrist strap when:
  - Removing a board from its antistatic bag
  - Connecting or disconnecting boards or peripherals

  The other end of the strap lead should be alternatively connected to:
  - a ground mat
  - grounded chassis metalwork
  - a facility electrical service ground
- Keep boards in the antistatic bags until they are needed
- Place circuit boards that are out of their antistatic bags on an antistatic mat if one is available. Such a mat should be grounded to a facility electrical service ground. Do not place boards on top of an antistatic bag unless the outside of the bag also has antistatic protective properties.
- Remove a board from its antistatic bag only when wearing a properly-connected ground strap.

## 5.3 Steps Before Installation

Pay attention to the following subsections before starting to install these boards. In addition, do the following:

1. Become familiar with the contents of the referenced documentation.
2. Verify that all listed hardware and software is available (see Section 5.1.1, *Order Items* ).

3. Check power and thermal requirements (see Section 5.3.1, *Check Power and Thermal Requirements*).
4. Verify that space, local area networking (LAN), and environmental preparations are completed (see Section 5.3.2, *Determine Local Network IP Addresses and Hostnames*).
5. Ensure that the hostnames and their network IP addresses are allocated and registered at the site.

## 5.3.1 Check Power and Thermal Requirements

Ensure that:

- Your enclosure specifications support the sum of the specified maximum board power loads. See Section 2.2.5, *Power Requirements* for board power specifications.
- Your enclosure specifications support the cooling airflow requirements. See Section 2.2.7, *Environmental Specifications*.
- Facility power loading specifications can support the rack or enclosure requirements

## 5.3.2 Determine Local Network IP Addresses and Hostnames

Collect the following information to connect hosts to the local area network (LAN). Ask your network administrator for help if necessary. This information is not needed for a standalone installation.

- IP addresses[1] and hostname for each Netra CP2060/CP2080 client
- Domain name
- Type of name service and corresponding name server names and IP addresses (for example DNS and NIS (or NIS+)
- Subnet mask
- Gateway router IP address
- NFS server names and IP addresses
- Web server URL

You may need the MAC (Ethernet) addresses of the local hosts to make nameserver database entries. The MAC address can be seen in the console output while booting to the OK prompt. It can also be derived from the Host ID seen on the label of the $I^2C$ EEPROM package.

1. Local IP addresses are not needed if they are assigned by a network DHCP server

## 5.4 Installation Procedure Summary

The procedure to setup and configure a Netra CP2060 or CP2080 board in a system includes the following steps:

1. Configure host board physical hardware. For example install memory, PMC cards, and set switches if necessary.
2. Configure the transition card with regard to PIM modules, switch settings, or connector attachments.
3. Physically install the host, transition card and any peripheral boards into the chassis.
4. Connect the host(s) to a local network. Alternatively, the host can be run as a standalone system without a network connection.
5. Install the operating system.

## 5.5 Configuring the Netra Board Hardware

This section lists the installation details for memory modules and PMC module/s.

### 5.5.1 Memory Module Installation (Netra CP2080 only)

Only the Netra CP2080 board can accommodate modular memory. For directions on the installation process of the memory module/s on the Netra CP2080, please refer to the document *Memory Module Installation/Removal Guide for CP2000 Family CompactPCI Boards* (P/N 816-0854-xx).

The Netra CP2060 board has only soldered memory on the motherboard and does not use memory modules.

### 5.5.2 PMC Module Installation

Use the PMC card manufacturer's procedure to install these cards.

## 5.6 Configuring Transition Card Hardware

If using the XCP2060-TRN I/O transition card, please refer to the *XCP2060-TRN I/O Transition Card Manual for Netra CP2060/CP2080 CompactPCI Boards* (P/N 806-6203-xx). You may also refer to this document for detailed connector pin assignments.

### 5.6.1 Installing PIM Assemblies

Use the PMC card manufacturer's procedure to install these cards. Only the PIM A and PIM B Connector Power Pin Assignments are provided in *Table 5-3*.

---

**Caution –** When installing a PIM card on to the transition card, please ensure that the PIM card power signals match the signals of the corresponding power signals of the PIM connectors that are to be installed on the XCP2060-TRN I/O Transition Card.

---

**Table 5-3** PIM A and PIM B Connector J0 Power Pin Assignments (J400 and J500)

| Pin | Signal | Pin | Signal | Pin | Signal | Pin | Signal |
|---|---|---|---|---|---|---|---|
| 1 | | 17 | | 33 | | 49 | |
| 2 | +12V | 18 | GND | 34 | GND | 50 | GND |
| 3 | | 19 | | 35 | | 51 | |
| 4 | | 20 | | 36 | | 52 | |
| 5 | +5V | 21 | +5V | 37 | +5V | 53 | +5V |
| 6 | | 22 | | 38 | | 54 | |
| 7 | | 23 | | 39 | | 55 | |
| 8 | | 24 | | 40 | | 56 | |
| 9 | | 25 | | 41 | | 57 | |
| 10 | +3.3V | 26 | +3.3V | 42 | +3.3V | 58 | +3.3V |
| 11 | | 27 | | 43 | | 59 | |
| 12 | | 28 | | 44 | | 60 | |
| 13 | GND | 29 | GND | 45 | GND | 61 | -12V |
| 14 | | 30 | | 46 | | 62 | |
| 15 | | 31 | | 47 | | 63 | |
| 16 | | 32 | | 48 | | 64 | |

# 5.7 Replacing the Serial EEPROM

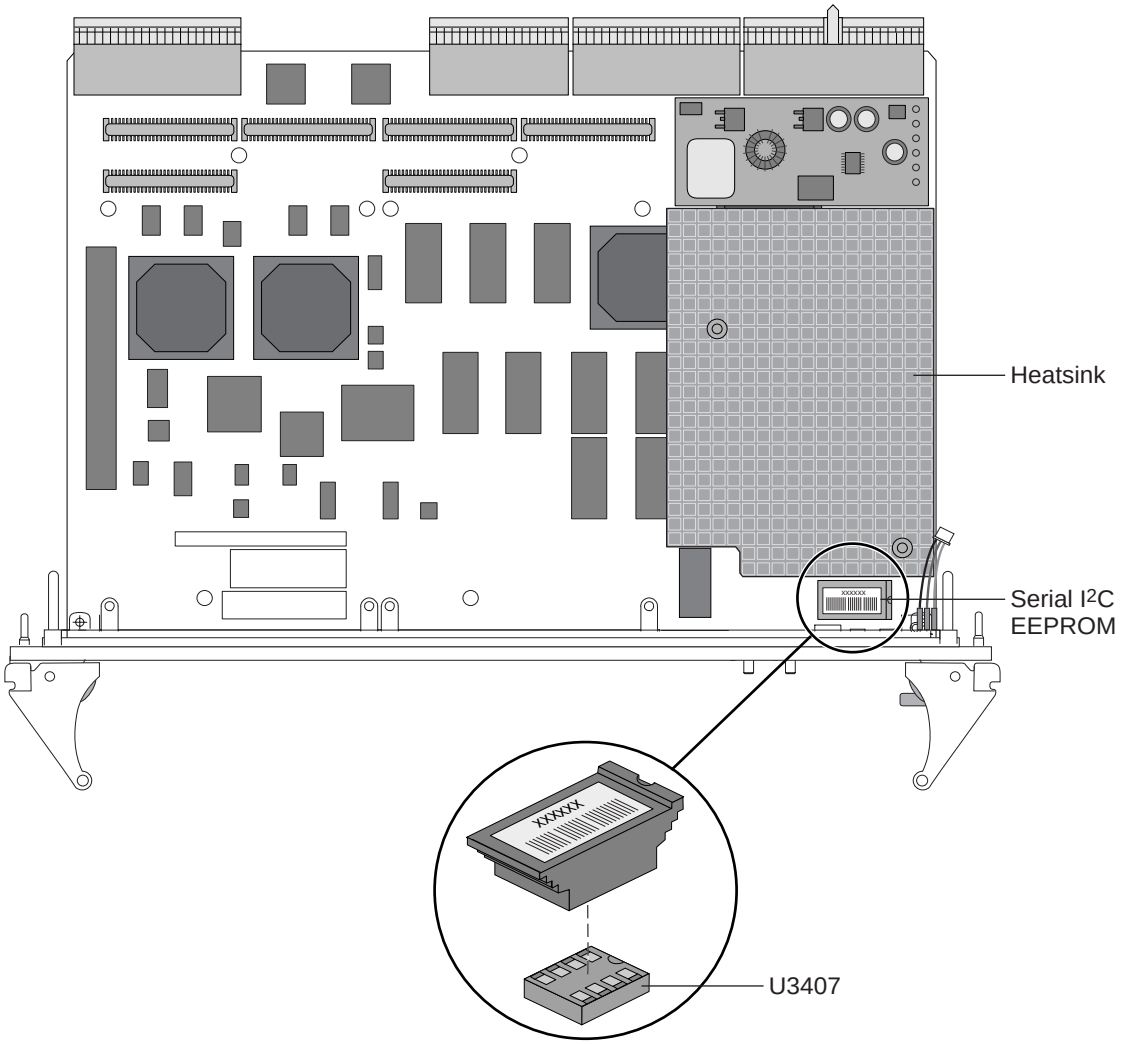



**Figure 5-5** Replacing the Serial I$^2$C EEPROM

The Serial I$^2$C EEPROM is the MAC address carrier and it stores the backup copy of the board MAC address and Host ID information (see Section 3.3.4.4, *Serial I2C EEPROM*).

The Netra CP2060/CP2080 board supports the Serial $I^2C$ EEPROM. If you need to replace the Netra CP2060/CP2080 board, remove the Serial $I^2C$ EEPROM from the original board and install it on the new Netra CP2060/CP2080 board.

To correctly position the host ID board and to precisely install it on the Netra CP2060/CP2080 board, see *Figure 5-5*.

## 5.8 Installing Boards into the CompactPCI Chassis

This section describes the installation of the transition card, the system host board, the satellite board and the IO cards into a Netra CP2060/CP2080 CompactPCI system chassis.

### 5.8.1 Installing the XCP2060-TRN IO Transition Card

Use the installation procedure detailed in the *XCP2060-TRN IO Transition Card Manual for Netra CP2060/CP2080 CompactPCI Boards* (P/N 806-6203-xx) to install the XCP2060-TRN Transition Card.

A compatible transition card is required to be used with the Netra CP2060 and CP2080 boards for I/O access. The transition card enables access to the network, to a boot device and to a console terminal. It is recommened that the OEM customer use the XCP2060-TRN IO Transition Card, but a customer may design their own transition card.

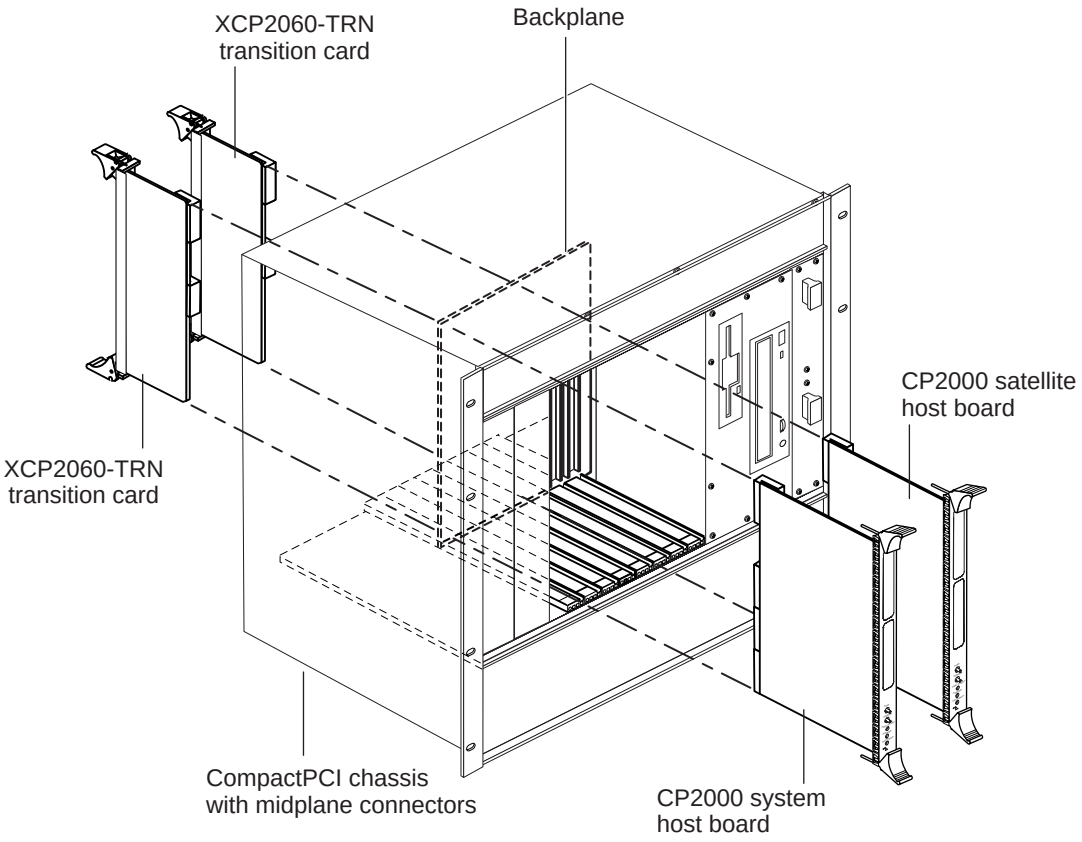



**Figure 5-6** Installing the XCP2060-TRN CPCI IO Transition Card

## 5.8.2 Installing the Netra CP2060/CP2080 Board

A CompactPCI chassis contains:

- A system slot, usually the leftmost (viewed from the front) whose position is indicated by a triangle symbol visible on the backplane (if the chassis meets the PICMG 2.0 CompactPCI specification).
- Seven peripheral slots (for a single-segment chassis). Peripheral slots are identified by a circle symbol visible on the backplane.

The Netra CP2060/CP2080 board may be used as a system controller or as a satellite board. Both, the Netra CP2060 and the Netra CP2080 boards can be installed in either a peripheral slot or in a system board slot. The Netra board's role is reliant on which slot in the CompactPCI chassis it is placed (see Section 3.4.3.1, *Arbitration in System Controller Role*).

1. **Ensure that power is disconnected from the chassis.**

   The Netra board can be installed while the chassis is powered—however *only start with a powered chassis if you must do so when inserting the board in the system host board slot.*

2. **Check that the corresponding XCP2060-TRN IO Transition Card is installed.**

   If you need a transition card for IO for the Netra board ensure that it is already present in the chassis. This step is essential if the chassis is powered during the installation. Take the step anyway as a point of safe procedure.

3. **Check positioning of the Netra board extraction levers.**

   Ensure that the Netra board extraction levers are aligned perpendicular to the card flange.

4. **Install the Netra board into the chassis front connector slot.**

   Install into the slot that is appropriate for the function of the Netra board.

   Position the Netra board such that its J5 connector—and the processor heatsink—are oriented upwards. Engage the board edges with the chassis card guides and slide it into the chassis. (*Figure 5-6*). Gently maneuver the card into registration with the keying hardware and the shoulders surrounding the backplane pins, without the card socket contacts engaging these pins. At this point, the card rear flange should project approximately 6 mm (1/4 in.) back from the "fully home" position. Apply pressure to engage the pins and seat the board. Slide the board into the top and bottom mounting rails and into the backplane while gently keeping the board handles pushed inwards and pushing the board into the chassis. See *Figure 5-7*.

5. **Install a screw through the top and bottom of the front connector plate to secure the board.**

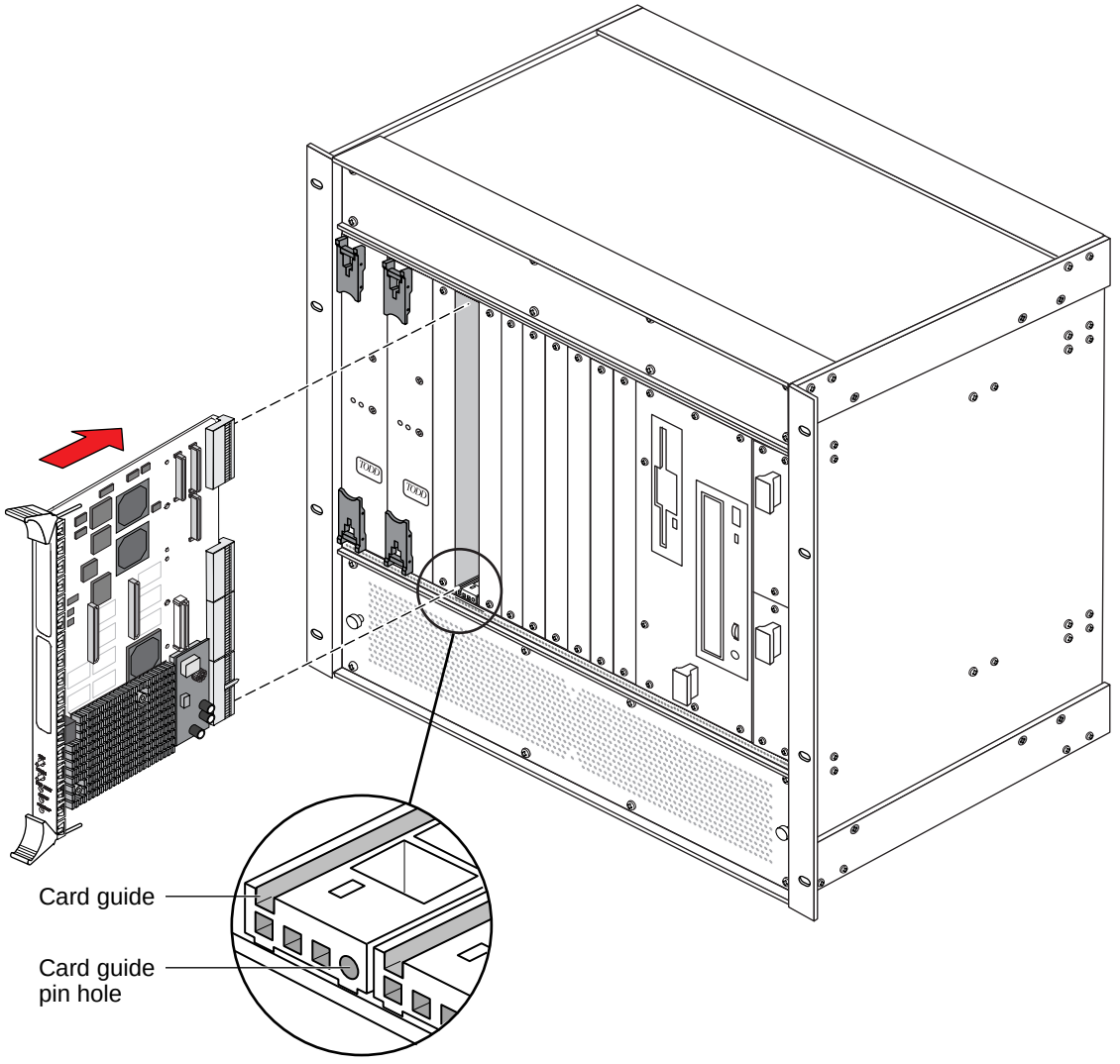



**Figure 5-7** Installing a Typical Netra CP2060 Board Into a CompactPCI Chassis in a System Host Slot

---

**Note –** Follow the CompactPCI chassis manufacturer's instructions to make sure the Netra CP2060/CP2080 board is installed into the system controller slot.

---

### 5.8.3 Installing an IO Board

Various IO boards (IO cards) can be installed in the chassis that contain the Netra board. If these boards meet Hot Swap requirements, these can be Hot Inserted and do not require the system to be powered down. These IO cards need to be installed in their respective slots. Refer to the chassis manufacturer's documentation for slot assignments for special installation instructions. The basic IO card installation procedure is the same as described in Section 5.8.2, *Installing the Netra CP2060/ CP2080 Board*.

The ejector handles on the IO cards may not be the same style as Netra board ejector handles. Some boards have retracting ejector handles. Refer to the IO card manufacture's instructions on how to engage or disengage ejector handles that secure that board to the chassis.

### 5.8.4 Attaching the Host(s) to a Local Network

Using a category 5 grade network cable, connect one RJ45 connector end into the receptacle on the rear flange of the transition card. The other end should be connected to a suitable 10/100 Mb Ethernet hub on the local subnet.

---

**Note –** The user needs to use shielded cables for Ethernet ports on the transition card and the shield should be grounded at both ends.

---

## 5.9 Bringing Up an Assembled Netra CP2060/CP2080 Board Computer

This section describes how to setup a computer that contains the Netra CP2060/ CP2080 board.

1. **Connect IO cabling to the host board and to the serial port of the host system.**
2. **Connect a serial cable to the ttya port of the XCP2060-TRN cPCI IO Transition Card of the target machine and to the serial port of the host machine.**
3. **Use the** `tip` **utility on the host system to establish a full-duplex terminal connection with the Netra CP2060/CP2080 board. At the UNIX prompt in a command tool or shell tool, type:**

   ```
   tip -9600 /dev/ttya
   ```

4. **Connect any other peripheral devices (such as a printer) to the appropriate connector.**

5. **Power on the system (that contains the Netra CP2060/CP2080 board) to run the power-on self-tests (POST). For details on POST, see Chapter 6,** *Firmware***.**

---

**Note –** This also depends on OBP environment variable settings (see Section 6.2.3, *OBP Configuration Variables*).

---

After running POST, install the Solaris Operating Environment package on the system that contains Netra CP2060/CP2080 (see Section 5.11, *Installing the Operating System*).

## 5.10 Setting Up a Diskless Environment

This section provides pointers on how to obtain information on building a remote boot server in a diskless environment. Thus, a Netra CP2060/CP2080 board acts as a diskless client booted over the network.

For information on how to set up a diskless server, see *5.10.1*. For information on how to use a tip line connection on a diskless client and boot over the net, see *5.10.2*.

### 5.10.1 How to Set up a Boot Server

For information on how to boot from a diskless server, please refer to the document *Solaris 8 Advanced Installation Guide* (P/N 806-0957-xx)

The document contains a section on *How to Create a Boot Server on a Subnet* and is available on line, in PDF format or for purchase through the following URL:

*http://docs.sun.com*

### 5.10.2 How to Boot as a Diskless Client

For information on working with diskless clients, please refer to the document *Solaris 8 System Administration Supplement* (P/N 806-6611-xx).

The document contains a section on *How to Set Up Your Diskless Client Environment* and *How to Add Diskless Clients* and is available on line, in PDF format or for purchase through the following URL:

*http://docs.sun.com*

## 5.11 Installing the Operating System

The Solaris 8 operating environment, Release 1/01 or future compatible versions may be used with the Netra CP2060/CP2080 board system. Refer to Solaris™ installation manuals for the installation procedure (see Chapter, *Bibliography*).

For additional functionality available through a CP2000 Supplemental CD, please refer to the product website at:

*http://www.sun.com/microelectronics/commprovider/cp2060/*

## 5.12 Configuring a Netra CP2060/CP2080 Satellite Board

Each Netra CP2060/CP2080 satellite board is capable of working in either basic or full hot swap modes. If the system is set to work in basic hot swap mode, then the board has to be manually configured. If the system is set to work in full hot swap mode, the process of configuration is automatic. Please refer to the READ ME file in the CP2000 Supplemental CD for details on setting the hot swap mode of the system. Information on obtaining the CD is available at the product website:

*http://www.sun.com/microelectronics/commprovider/cp2060/*

When inserted in a cPCI chassis, a satellite board needs to be configured before it can be used. This process ensures that a board is “recognized” by the system (see also Section 5.13, *Hot Swapping Process*). For information on configuring an IO board, see Section 5.14, *Configuring a Peripheral IO Board*.

You can check the current status of all the cards in the system by executing the following command in the superuser mode from the system board:

# **cfgadm**

This will display a screen showing the current status of the Hot Swappable system components

```
Ap_Id              Type           Receptacle     Occupant   Condition
c0                 scsi-bus       connected      configured unknown
c1                 scsi-bus       connected      unconfiguredunknown
pci1:hsc0_slot2    unknown        disconnected   unconfiguredunknown
pci1:hsc0_slot3    unknown        disconnected   unconfiguredunknown
pci1:hsc0_slot4    unknown        disconnected   unconfiguredunknown
pci1:hsc0_slot5    stpcipci/fhs   connected      configured ok
pci1:hsc0_slot6    unknown        disconnected   unconfigured unknown
pci1:hsc0_slot7    unknown        disconnected   unconfigured unknown
pci1:hsc0_slot8    pci-pci/bhs    connected      configured   ok
```

1. **Enter the following command as root from the system board to configure a board in slot 4:**

   **# cfgadm -c configure <**pci1:hsc0_slot4**>**

---

**Note –** The slot number shown in angle brackets may vary depending upon the location of the board in the chassis. Change the slot number accordingly.

---

Wait until the satellite board configuration is complete. You may confirm configuration on the display screen, with the **cfgadm** command. Once the satellite board is configured, you can boot the board from its own console (see also *cfgadm* man page):

To extract a basic hot swappable satellite board, check to see if any interface is running on the satellite board.

1. **Execute the following command from the system board:**

   **# ifconfig -a**

   If an interface is running then it will show up on the display screen.

2. **Execute the command to bring down the operating system from the satellite board:**

   **# uadmin 2 0**

3. **To unconfigure the satellite board type from the system board:**

   **`# cfgadm -c unconfig <board AP ID>`**

   (for example the IO board AP ID in this case is **`<pci0:hsc0_slot4>)`**

4. **Wait for the blue LED on the satellite board to turn ON before extracting it from the chassis.**

# 5.13 Hot Swapping Process

If the Netra CP2060/CP2080 board is being used as a system controller board and not functioning as a satellite board, it cannot be Hot Swapped at this time. The information provided below is useful in understanding the CP2060/CP2080 satellite board and IO board Hot Swap.

## 5.13.1 Status LED

The Hot Swap blue LED, located on the front panel of the Netra CP2060/CP2080 board (see *Figure 3-20* for location), is lit when it is permissible to insert or extract a board. This LED indicates that the system software has been placed in a state for orderly extraction of the board. Upon insertion of a board in a CompactPCI system, the LED is lit automatically until the hardware connection process completes. The LED then remains off until used by software to indicate that extraction is once again permitted.

---

**Note –** The user should wait for the blue LED light to turn OFF on the last inserted board, before inserting or extracting another board.

---

In the case of a Netra CP2060/CP2080 satellite board, during the insertion process when a board is hot inserted, the blue LED is lit again automatically by the system hardware until the hardware connection process is complete. The blue LED is turned OFF by the software until it indicates once again that the extraction of board is permitted.

Two ejector handles are used for inserting and extracting a CompactPCI board. A lower ejector handle is tied to a switch that turns the blue LED ON when you try to unlock the board for extraction. This switch changes its state once the ejector handle is pressed or unlocked, but before the movement of the board begins. You need to wait until the blue LED is ON before extracting the board.

## 5.13.2 Satellite Board and IO Board Hot Swap Process

The hot swap process for Netra CP2060/CP2080 satellite boards and IO boards in Basic Hot Swap mode and Full Hot Swap mode are described in the following two sections (also see Section 5.12, *Configuring a Netra CP2060/CP2080 Satellite Board* and Section 5.14, *Configuring a Peripheral IO Board*):

- Basic Hot Swap Process
- Full Hot Swap Process

### 5.13.2.1 Basic Hot Swap Process

The Basic Hot Swap is initiated by executing the `cfgadm` configure/unconfigure administration command. The `cfgadm` command provides configuration administration operations on dynamically reconfigurable hardware resources. Refer to the man pages (by entering % `man cfgadm`) for additional information on the options associated with this command.

1. **Execute the unconfiguration command from the system board** :

   **`# cfgadm -c unconfigure`** *<pci1:hsc0_slot5>*

   It informs the system to perform the process of extraction and insertion of a device, if permissible. The blue LED is turned ON if the process is successful, indicating that the board can be extracted.

2. **If hot swapping an IO board, go directly to Step 3. If hot swapping a satellite board , type this additional step from the satellite board**:

   **`# uadmin 2 0`**

3. **Unlock both the ejector handles of the board by pressing the red buttons. Wait for the blue light to turn ON. Do no unseat the board until the target board LEDs on the system status panel are in the proper state.**

   Even if you now decide that you do not want to remove the board, you must completely unseat the board, reset the ejection levers again in order to start the software driver attachment process. Simply locking the ejection levers after unlocking them will not start the driver attachment process.

---

**Caution –** Do not deactivate any other boards until the previous board has been deactivated. Deactivating more than one board at the same time can lead to unpredictable results.

---

4. **Extract the board and set it aside.**

5. **Slide the new board into the top and bottom mounting rails in the same slot into the backplane while gently keeping the board handles in open position. Once the board is all the way in, lock the ejector handles (see** *FIGURE 8-4***). The blue LED will be lit once the board makes complete contact with the backplane. The LED will go OFF shortly after the ejector handles are closed, if the hardware initialization of the board is successful. Otherwise, it will remain ON, indicating that the board has to be replaced.**

6. **Wait for the blue LED of the board to go OFF.**

7. **Execute the configuration command from the system board**:

   **`# cfgadm -c configure`** *<pci1:hsc0_slot5>*

8. **Take any other step necessary for board-specific configuration.**

### 5.13.2.2 Full Hot Swap Process

Full Hot Swapping a board does not require the system power to be turned OFF. Refer to the chassis manufacturer's documentation for slot assignments and additional information. In addition, a Full Hot Swap board does not require running the `cfgadm` administrative command. The system automatically configures the board for using the system resources. The board-specific configuration steps are still necessary after the completion of host-specific configuration depending on whether it is a satellite board or an IO board (see Section 5.12, *Configuring a Netra CP2060/ CP2080 Satellite Board* and Section 5.14, *Configuring a Peripheral IO Board*.)

Follow these steps to Hot Swap a Netra CP2060/CP2080 board or an IO board in a chassis that contains the Netra CP2060/CP2080 board system host board.

1. **Unlock both the ejector handles of the satellite/IO board by pressing the red buttons. Wait for the blue light to turn ON. Do no unseat the board until the target board LEDs on the system status panel are in the proper state.**

   Even if you now decide that you do not want to remove the board, you must completely unseat the board, reset the ejection levers again in order to start the software driver attachment process. Simply locking the ejection levers after unlocking them will not start the driver attachment process.

2. **Extract the IO board and set it aside.**

3. **Slide the new satellite/IO board into the top and bottom mounting rails in the same slot into the backplane while gently keeping the board handles in open position. Once the board is all the way in, lock the ejector handles (see** *Figure 5-8***). The blue LED will be ON once the board makes complete contact with the backplane. The LED will go OFF shortly after the ejector handles are closed, if the hardware initialization of the board is successful. Otherwise, it will remain ON, indicating that the board has to be replaced.**

4. **Once the blue LED of the board is OFF, then secure the board to the chassis.**

---

**Caution –** Do not deactivate any other boards until the first board has been deactivated. Deactivating more than one board at the same time can lead to unpredictable results.

---

*Figure 5-8* shows how to release the Netra CP2060/CP2080 injector/ejector handles.

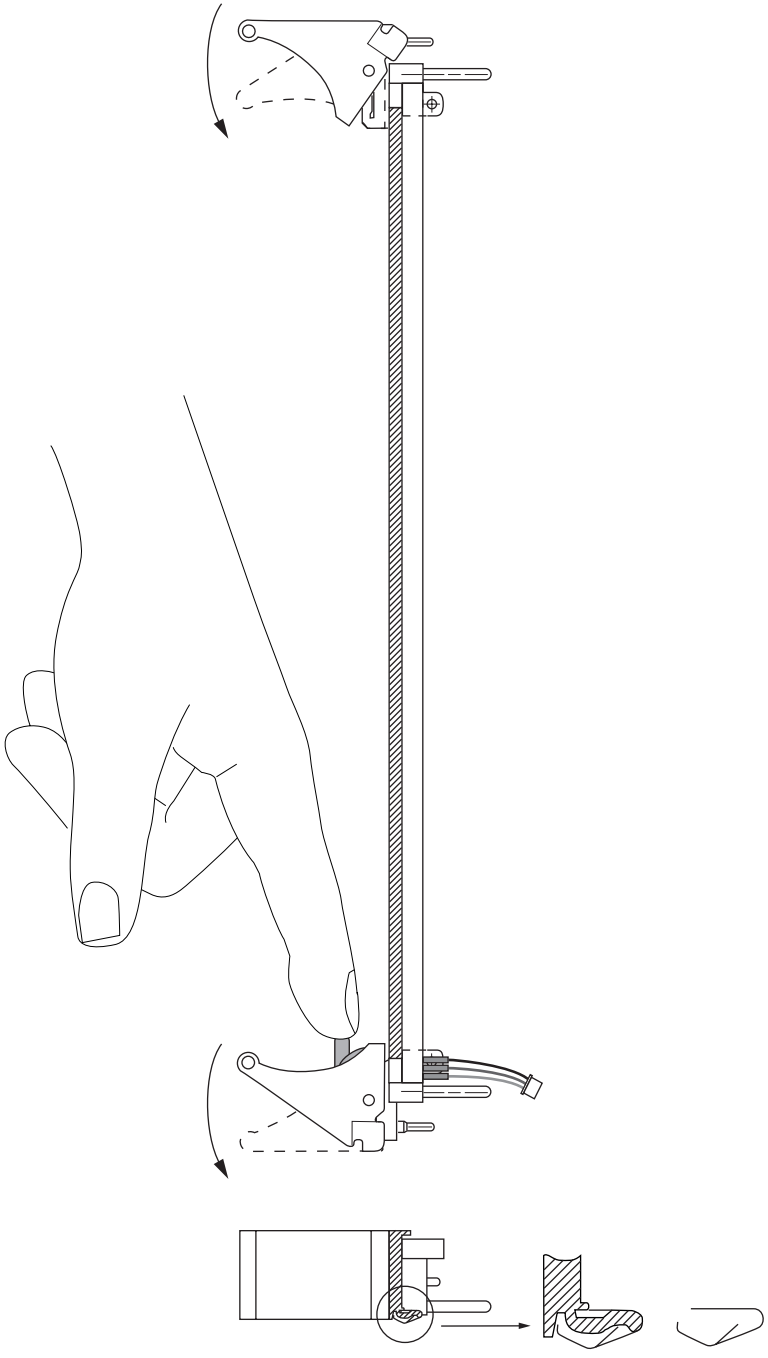


**Figure 5-8** Releasing the Netra CP2060/CP2080 Injector/Ejector Handles

# 5.14 Configuring a Peripheral IO Board

Each basic hot swappable IO board, when inserted in a chassis with Netra CP2060/CP2080, needs to be configured before it can be used. This process ensures that a board is "recognized" by the system (see also Section 5.13, *Hot Swapping Process*). For information on configuring a satellite board, see Section 5.12, *Configuring a Netra CP2060/CP2080 Satellite Board*.

---

**Note –** No manual configuration is required to configure an IO board for full hot swap mode. It is done automatically upon insertion.

---

You can check the current status of all the cards in the system by executing the following command in the superuser mode from the system board:

# **cfgadm**

This will display a screen showing the current status of the Hot Swappable system components like this:

```
Ap_Id              Type           Receptacle     Occupant    Condition
c0                 scsi-bus       connected      configured  unknown
c1                 scsi-bus       connected      unconfiguredunknown
pci1:hsc0_slot2    unknown        disconnected   unconfiguredunknown
pci1:hsc0_slot3    unknown        disconnected   unconfiguredunknown
pci1:hsc0_slot4    unknown        disconnected   unconfiguredunknown
pci1:hsc0_slot5    stpcipci/fhs   connected      configured  ok
pci1:hsc0_slot6    unknown        disconnected   unconfigured unknown
pci1:hsc0_slot7    unknown        disconnected   unconfigured unknown
pci1:hsc0_slot8    pci-pci/bhs    connected      configured   ok
```

1. **Enter the following command from the system board as root to configure a board in slot 4:**

   **# cfgadm -c configure <**pci1:hsc0_slot4**>**

---

**Note –** The slot number shown in angle brackets may vary depending upon the location of the board in the chassis. Change the slot number accordingly.

---

The board configuration is complete. You may confirm configuration on the display screen, with the **cfgadm** command (see also *cfgadm* man page for further details):

2. **Execute the following commands from the system board if you are going to configure a networking board with qfe (quad fast Ethernet) interfaces. For example, configure qfe0, which is assumed to be assigned to the first physical Ethernet interface of the network board and is the IO board network interface.**

   ```
   # ifconfig <qfe0> plumb
   # ifconfig <qfe0> <hostname> up
   ```

   To extract a basic hot swappable IO board, first check to see if any interface is running on the IO board.

1. **Execute the following command from system board:**

   ```
   # ifconfig -a
   ```

   If an interface is running then it will show up on the display screen.

2. **To close the interface, type from system board:**

   ```
   # ifconfig <qfe0> unplumb
   ```

3. **To unconfigure the IO board type from system board:**

   ```
   # cfgadm -c unconfig <IO board AP ID>
   ```

   (for example the IO board AP ID in this case is **<**`pci0:hsc0_slot4`**>**

4. **Wait for the blue LED on the IO board to turn ON before extracting it from the chassis.**

CHAPTER 6

# Firmware

The Netra CP2060/CP2080 board platform comprises a modular firmware architecture that gives the user latitude in controlling boot initialization. This latitude can be exploited in user customization of initialization and test firmware, even enabling installation of a custom operating system.

This chapter describes the structure and function of initialization firmware. For a more detailed description of the diagnostic parts, see Section 6.10, *Firmware Diagnostics*.

This platform also employs the System Management Controller (SMC) — described in Section 3.6, whose operation controls the CompactPCI interface, System Management and Hot Swap control, and some board hardware. The SMC configuration is controlled by separate firmware.

## 6.1 Initialization Firmware

Control flow at board startup is shown in *Figure 6-1*. Execution begins in Firmware Common Operations & Reset Environment (CORE)—which includes Basic POST (BPOST). It passes to Comprehensive POST (CPOST) and Extended POST (EPOST), if these are present, before returning to firmware CORE and on to OBP.

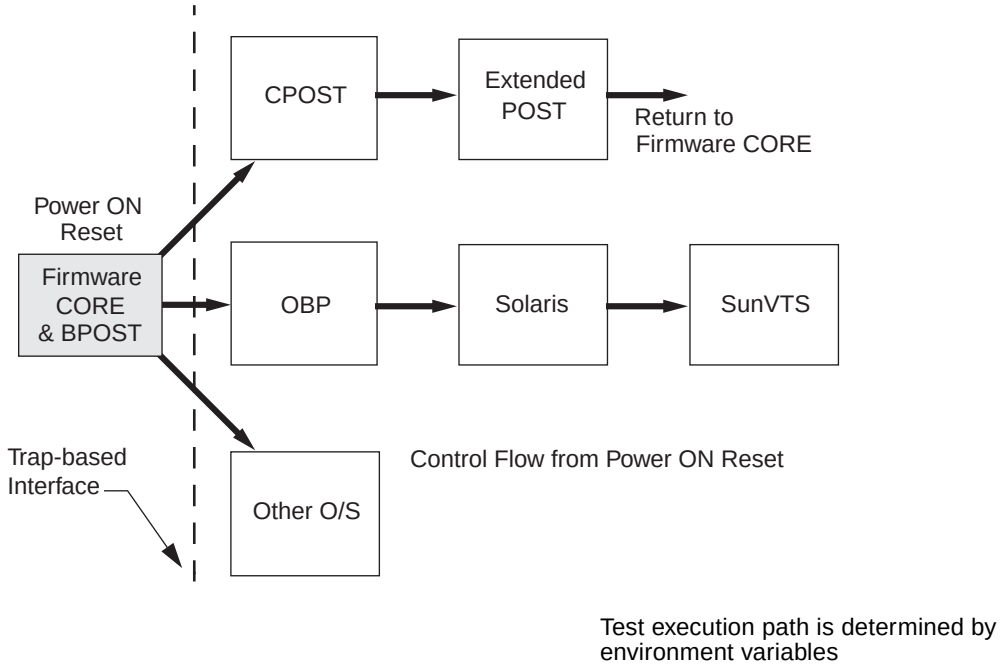



**Figure 6-1** Control Flow from Power On for Firmware CORE and Client Modules—Solaris Case

## 6.1.1 Firmware CORE and BPOST

Firmware CORE:

- Unifies system initialization and IO Operations for a higher level client, for example, OBP for Solaris software
- Avoids any duplication of effort for the same type of functions among various clients
- Provides a unified interface to higher level software using a soft trap mechanism. Trap services (software interrupts) are used to abstract hardware-dependent features behind a uniform service interface. Sun SPARC processors are designed with a common software trap structure that is useful for this common programming interface, so that clients may not need to carry another copy of those drivers and may use those services provided by Firmware CORE until their driver takes over.
- Provides access, early in the boot sequence, to the hardware-dependent services needed for client initialization; examples are IO devices including serial port and net.

- Provides basic system tests that can replace existing POST in *min mode*
- Enables extensive system testing to be done using the POST dropin in *max mode*
- Provides error recovery from exceptions which currently does not exist in OBP
- Enables use of popular languages with efficient compilation and easier debugging for development

BPOST is integrated into Firmware CORE. Its tests are interleaved with the initialization activities of Firmware CORE so as to present a foundation of validated and initialized hardware to run subsequent code such as that in CPOST or OBP. The tests listed in *Table 6-1* are examples of CORE and BPOST flow of execution.

---

**Note –** Not all of the hardware listed in this table is present on this platform. When a hardware item is not detected by the firmware, this firmware simply makes no attempt to test or initialize it.

---

Because BPOST runs from PROM, its extent of testing is limited to that needed by modules that are loaded later. Such a module, for example CPOST, can perform comprehensive testing more quickly because it executes from DRAM.

**Table 6-1** Firmware CORE and BPOST Flow of Execution

| Firmware CORE Service | Detail |
|---|---|
| Initialize Processor | sets processor in stable state |
| Initialize NVRAM | sets up state variables |
| Initialize E-bus and bridges | initializes E-bus and UPA/PCI and PCI/PCI bridges in path between CPU & E-bus devices |
| InitializeTTY | for message display |
| Set memory timings | |
| Verify NVRAM | check magic number. Set defaults if bad |
| Check keyboard | probe & initialize keyboard, set TTYA otherwise |
| Check I/P device for key pressed | set state variables in NVRAM accordingly |
| Cache, MMU test | perform basic diagnostics on caches & MMUs[1] |
| Initialize caches, MMUs | setup I and D caches and MMUs |
| Memory test | perform partial memory test[2] |
| Memory probe | probe memory & clear top memory region |
| MMU and cache setup | setup I/D MMUs with valid mappings; enable MMUs and I/D caches |
| Copy Firmware CORE | copy Firmware CORE into memory and transfer control to the RAM copy |
| Setup trap table | setup trap table in memory |
| Initialize interrupts | set up hardware interrupts |
| Initialize TOD | |
| Set up CPU counter | calibrate CPU counter to determine module speed |
| Probe PCI bus | probe for Primary PCI system bus |
| Execute POST dropin[2] | |
| Locate the client | locate the client in PROM. If found, copy into memory & transfer control to it |
| Enter user interface | OBP for Solaris software, else RTOS or custom OS |

1. Execute if hardware power-on, run-post set to true, post-level set to min/max and key to skip post not pressed
2. Execute if h/w power-on, run-post set to true, post-level set to max and key to skip post not pressed

### 6.1.2 CPOST and EPOST

CPOST contains tests for higher level board functions. By placing these tests in a separate module, the user has the option of performing them and the developer can substitute them with other tests. Examples of CPOST tests are:

- PCI configuration register test for PCIO-2
- DMA tests

### 6.1.3 EPOST

EPOST is used for additional POST code dropins that are provided by the user.

### 6.1.4 OBP

Rather than executing the initialization code that formerly existed in OBP for prior Sun board platforms, OBP now makes calls to the traps laid down by Firmware CORE. OBP exists in the form of a dropin in the System Flash memory area.

OBP probes for devices and builds the device tree, which is a table that contains entries for how drivers communicate with connected hardware. Each line, or entry, of the device tree is a reference for the node entry for the peripheral in the /dev directory in the / directory. The device tree is inherited by Solaris software as it is booted. An example of a device tree is shown below. The device tree can be seen by typing: **show-devs** at the *ok* prompt. An example of a device tree appears below.

**Table 6-2** Example of a show-devs Device Tree

```
ok show-devs
/SUNW,UltraSPARC-IIe@0,0
/pci@1f,0
/virtual-memory
/memory@0,0
/aliases
/options
/openprom
/chosen
/packages
/pci@1f,0/pci@1
/pci@1f,0/pci@1,1
/pci@1f,0/pci@1/pci@1
/pci@1f,0/pci@1,1/usb@3,3
/pci@1f,0/pci@1,1/network@3,1
/pci@1f,0/pci@1,1/usb@1,3
/pci@1f,0/pci@1,1/network@1,1
/pci@1f,0/pci@1,1/ebus@3
/pci@1f,0/pci@1,1/ebus@1
/pci@1f,0/pci@1,1/usb@3,3/device@3
/pci@1f,0/pci@1,1/ebus@3/sysmgmt@14,600000
/pci@1f,0/pci@1,1/ebus@1/flashprom@10,400000
/pci@1f,0/pci@1,1/ebus@1/su@14,300000
/pci@1f,0/pci@1,1/ebus@1/su@14,320010
/pci@1f,0/pci@1,1/ebus@1/idprom
/pci@1f,0/pci@1,1/ebus@1/eeprom@14,0
/pci@1f,0/pci@1,1/ebus@1/flashprom@10,0
/openprom/client-services
/packages/kbd-translator
/packages/dropins
/packages/SUNW,builtin-drivers
/packages/disk-label
/packages/obp-tftp
/packages/deblocker
/packages/terminal-emulator
ok
```

OBP also contains aliases for some of the devices shown in the device tree, These aliases can simplify hardware access at the ok prompt, for example:

ok **boot disk1**

```
ok devalias

userprom1            /pci@1f,0/pci@1,1/ebus@1/flashprom@10,400000
hsc                  /pci@1f,0/pci@1,1/ebus@3/sysmgmt@14,600000
dload                /pci@1f,0/pci@1,1/network@1,1:,
systemprom           /pci@1f,0/pci@1,1/ebus@1/flashprom@10,0
pcic                 /pci@1f,0/pci@1/pci@1
pcib                 /pci@1f,0/pci@1,1
pcia                 /pci@1f,0/pci@1
ebus2                /pci@1f,0/pci@1,1/ebus@3
ebus                 /pci@1f,0/pci@1,1/ebus@1
net2                 /pci@1f,0/pci@1,1/network@3,1
net                  /pci@1f,0/pci@1,1/network@1,1
ttya                 /pci@1f,0/pci@1,1/ebus@1/su@14,320010
ttyb                 /pci@1f,0/pci@1,1/ebus@1/su@14,300000
ok

hsc                  /pci@1f,0/pci@1,1/ebus@3/sysmgmt@14,600000
dload                /pci@1f,0/pci@1,1/network@1,1:,
userprom1            /pci@1f,0/pci@1,1/ebus@1/flashprom@10,400000
systemprom           /pci@1f,0/pci@1,1/ebus@1/flashprom@10,0
pcic                 /pci@1f,0/pci@1/pci@1
pcib                 /pci@1f,0/pci@1,1
pcia                 /pci@1f,0/pci@1
ebus2                /pci@1f,0/pci@1,1/ebus@3
ebus                 /pci@1f,0/pci@1,1/ebus@1
net2                 /pci@1f,0/pci@1,1/network@3,1
net                  /pci@1f,0/pci@1,1/network@1,1
ttya                 /pci@1f,0/pci@1,1/ebus@1/su@14,320010
ttyb                 /pci@1f,0/pci@1,1/ebus@1/su@14,300000
name                 aliases
```

## 6.2 Firmware NVRAM Variables

This section provides some information on the CORE NVRAM variables and the NVRAM configuration variables.

## 6.2.1 Firmware CORE NVRAM Variables

At start up, Firmware CORE defines a set of variables in the NVRAM. These provide for controlling initialization and selecting the amount of testing required. These variables determine the following functions. At the CORE interface, type `print-nvram` and the fixed offset NVRAM variables *similar* to the following will be displayed on the screen (see *Table 6-4*):

```
user-interface = 0
run-post       = ff
post-level     = 40
kernel         = FVM
trap-state     = 60
msg-verbosity  = 3
```

## 6.2.2 Firmware CORE Execution Control

The key combinations listed in *Table 6-3* can be used to control the flow of execution at system boot. These key combinations must be pressed at Power-on.

**Table 6-3** Key Sequences

| Key combination | Result |
|---|---|
| <Control><P> | skip POST |
| <Control><U> | enter CORE user interface |
| <Control><N> | set default NVRAM variables |
| <Control><M> | turn on power on messages |

The Netra CP2060/CP2080 board supports the USB keyboard.

## 6.2.3 OBP Configuration Variables

Configuration variables are used by the OBP code and are stored in NVRAM. The following is a sample of the output when the `printenv` command is entered at the `ok` prompt. The `setenv` command is used to modify the environment variables. The boot process is controlled by several variables. See *Table 6-5*. For values of each variable, refer to the *OpenBoot 3.x Command Reference Manual* (see Bibliography).

**Table 6-4** NVRAM Configuration Variables

| Parameter | Value | Default Value | Description |
|---|---|---|---|
| env-monitor | disabled | disabled | environment monitoring at OBP (enabled or disabled). |
| warning-temperature | 60 | 60 | sets the CPU warning temperature |
| shutdown-temperature | 65 | 65 | sets the CPU shutdown temperature |
| diag-passes | 1 | 1 | |
| diag-continue? | 0 | 0 | |
| diag-targets | 4 | 0 | |
| diag-verbosity | 3 | 0 | |
| keyboard-click? | false | false | if true, enable keyboard click |
| keymap | | | key map for custom keyboard |
| scsi-initiator-id | 7 | 7 | |
| #power-cycles | 1431655901 | no default | initialized in manufacture |
| system-board-serial# | | no default | initialized in manufacture |
| system-board-date | | no default | initialized in manufacture |
| ttyb-rts-dtr-off | false | false | if true, OS does not assert DTR and runs on TTYB |
| ttyb-ignore-cd | true | true | if true, OS ignores TTYB carrier-detect |
| ttya-rts-dtr-off | false | false | if true, OS does not assert DTR and runs on TTYA |
| ttya-ignore-cd | true | true | if true, OS ignores TTYA carrier-detect |
| ttyb-mode | 9600,8,n,1,- | 9600,8,n,1,- | TTYB (baud, #bits, parity, #stop, handshake) |
| ttya-mode | 9600,8,n,1,- | 9600,8,n,1,- | TTYA (baud, #bits, parity, #stop, handshake) |
| enable-netconsole? | false | false | |
| cpci-probe-list | 0,1,2,3,4,5,6,7,8,9,a,b,.. | 0,1,2,3,4,5,6,7,8,9,a,b,.. | probe list for devices present on cPCI bus |
| pcia-probe-list | 1 | 1 | probe list for devices present on internal PCI bus A |
| pcib-probe-list | 1,2,3,4 | 1,2,3,4 | probe list for devices present on internal PCI bus B |
| mfg-mode | off | off | manufacturing test mode (leave off) |

**Table 6-4** NVRAM Configuration Variables

| Parameter | Value | Default Value | Description |
|---|---|---|---|
| diag-level | max | max | level of diagnostics to run (min or max) |
| watchdog-timeout | 65535 | 65535 | |
| watchdog-enable? | false | false | |
| fcode-debug? | false | false | |
| output-device | screen | screen | console output device (usually screen, ttya or ttyb) |
| input-device | keyboard | keyboard | console input device (usually screen, ttya or ttyb) |
| load-base | 16384 | 16384 | base address where client image is loaded by OBP |
| auto-boot-retry? | false | false | |
| boot-command | boot | boot | command that is executed if auto-boot? is true |
| auto-boot? | false | true | if true, boot automatically after power-on reset |
| watchdog-reboot? | false | false | if true, reboot after watchdog reset |
| diag-file | | | file from which to boot in diagnostic mode |
| diag-device | net | net | device from which to boot |
| boot-file | | | file to boot (an empty string lets secondary boot choose default) |
| boot-device | disk net | disk net | device from which to boot |
| local-mac-address? | false | false | if true, local-mac-address is used; else, whwtem-wide mac-address is used when booting over any network interface in this system |
| net-timeout | 0 | 0 | |
| ansi-terminal? | true | true | not applicable |
| screen-#columns | 80 | 80 | screen width in columns |
| screen-#rows | 34 | 34 | screen height in rows |
| silent-mode? | false | false | if true, suppress messages related to clearing memory |
| use-nvramrc? | false | false | if true, execute commands in NVRAMRC during system start-up |
| nvramrc | ”ebus2” select-dev 000... | | contents of NVRAMRC |

**Table 6-4** NVRAM Configuration Variables

| Parameter | Value | Default Value | Description |
|---|---|---|---|
| security-mode | none | no default | firmware security level (none, command or full)<br>none: no password required (default)<br>command: all commands except for `boot` and `go` require password<br>full: all commands except for go require password |
| security-password | | no default | firmware security password (never displayed) |
| security-#badlogins | 0 | no default | OBP internal use |
| oem-logo | | no default | byte array custom OEM logo (enabled by oem logo? true); displayed in hex |
| oem-logo? | false | false | if true, use custom OEM logo, else use Sun logo |
| oem-banner | | no default | custom OEM banner (enabled by oem-logo? true) |
| oem-banner? | false | false | if true, use custom OEM banner |
| hardware-revision | UUUUUUUU... | no default | initialized in manufacture |
| last-hardware-update | UUUUUUUU... | no default | initialized in manufactture |
| diag-switch? | true | false | if true, POST is executed when system is next powered |

**Note –** All numbers are considered as HEX numbers.

*The diag- switch* and *diag- level* variables listed in *Table 6-4* affect the path through the various embedded tests. *Table 6-5* shows the effect of setting these variables.

BPOST is embedded within Firmware CORE and is executed when the OBP environment variable, *diag- switch* is set to *true* and *diag- level* set to *min*. Similarly CPOST (and EPOST if it is present) is executed when diag- level is set to *max*. The permutations are shown in *Table 6-5*.

**Table 6-5** OBP Environment Variable Settings for Executing the POST Modules

| Module | diag-switch?[1] set: | diag-level[1] set: | Description |
|---|---|---|---|
| | false | X | no messages are output to TTY |
| BPOST | true | min (0x20) | |
| | true | off (0x0) | messages are output to TTY |
| | false | X | no messages are output to TTY |
| CPOST | true | max (0x40) | runs after BPOST |
| | true | off (0x0) | messages are output to TTY |
| | false | X | no messages are output to TTY |
| EPOST | true | max (0x40) | runs automatically after CPOST (if EPOST module is present) |
| | true | off (0x0) | messages are output to TTY |

1. Firmware CORE variables *run-post* and *post-level* are equivalent to env. variables diag-switch? and diag-level respectively.

# 6.3 Firmware Memory Map

The satellite host board boots from the 1MB system flash PROM device which contains the Firmware CORE, Basic POST code, Comprehensive POST, and OBP. The contents map of this PROM is shown in *Figure 6-2*. User-developed code can also be programmed into the user flash memory space in the form of *dropins*. The system flash may be upgraded by running a program out of OBP—see Section 6.8, *Field CORE/OBP Firmware Upgrade*. It is not otherwise accessible by the user.

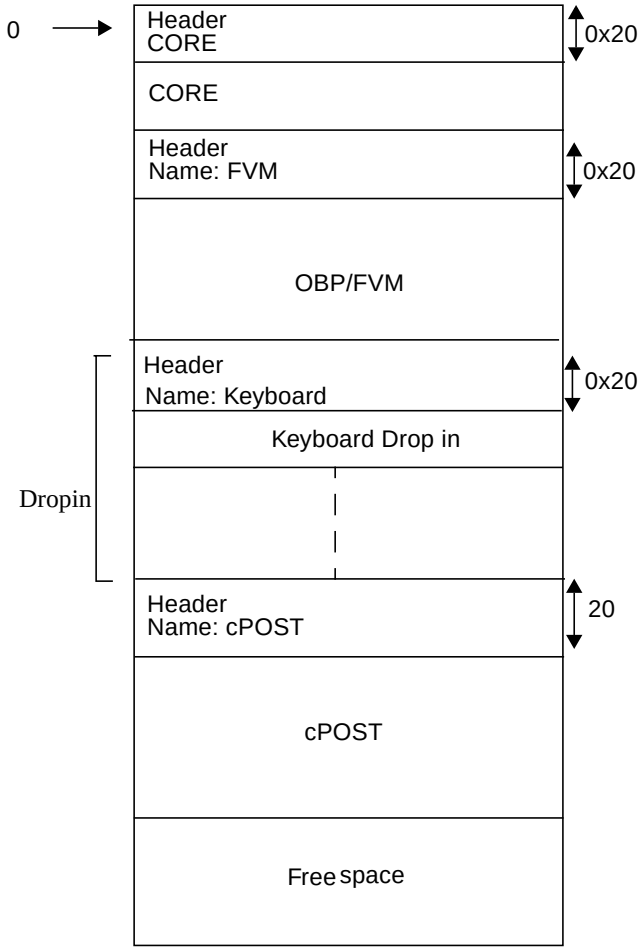



**Figure 6-2** System Flash PROM Map

# 6.4 Firmware CORE Features

*Table 6-6* lists the firmware CORE Commands that are run from the monitor. At the key sequence <Control><U> mode (see *Table 6-3*), you may type help to get all the supported commands such as in the example shown below.

**Table 6-6** Monitor Commands CORE

| Description of Task | CORE Monitor Command |
|---|---|
| To get this help | help |
| To allocate memory buffer | malloc <size> |
| To free memory buffer | free <addr> |
| To block copy memory | bcopy <src> <dest> <#bytes> |
| To dump memory | dump <addr> <#bytes> [asi] |
| To read an address | [safe-]peek <addr> <1\|2\|4\|8> [asi] |
| To write to an address | poke <addr> <1\|2\|4\|8> <data> [asi] |
| To update Flash PROM | flash-update <dev> <file-path> |
| To load a file | load <device> <file-path> <addr> |
| Jump to an address | go <addr> |
| Execute client | execute [client-name] |
| Print NVRAM data | print-nvram |
| Write to NVRAM variable | set-nvram <variable-name\|ID> <data> |
| Read an NVRAM variable | get-nvram <variable-name\|ID> |
| Delete an NVRAM variable | delete-nvram <ID> |
| Set NVRAM vars to default | set-defaults |
| Call a trap function | trap <trap#> <par0> ... <par5> |
| Soft Reset | reset |
| To change input device | input-device <tty\|kbd> |
| To initialize PCI | init-pci |
| To show all pci devices | show-pci-devs |
| To show pci config space | show-pci-space <bus#> <device#> <function#> <offset> |
| To show pci nexus nodes | show-nexus-nodes |
| To remove a pci device | rm-pci-dev <device#> |

**Table 6-6** Monitor Commands CORE

| Description of Task | CORE Monitor Command |
|---|---|
| To add a pci device | add-pci-dev <device#> |
| To remove all pci devices | rm-pci-devs |
| To add all pci devices | add-pci-devs |
| To execute UI cmd in loop | loop <count> <command> |

**Note –** All numbers are considered as HEX numbers

# 6.5 USB Keyboard Support

The Netra CP2060/CP2080 board supports USB keyboard only.

# 6.6 ASM Support at OBP

Advanced System Monitoring (ASM) is an intelligent fault detection system to increase uptime and manageability at OBP. The SMC module on the Netra CP2060/CP2080 board, supports the temperature monitoring functions of ASM. ASM monitors the following at regular intervals at the ok prompt:

- CPU heatsink thermal sensor
- ENUM signal on the system host board
- PCI_RESET# polling on the satellite board

## 6.6.1 CPU Heatsink Thermal Sensor

At the OBP level, when an over-temperature condition occurs, corresponding messages are displayed on the console. OBP displays the warning messages as soon as the board temperature reaches the warning temperature and is still below the shutdown temperature. The shutdown messages are displayed as soon as the board temperature reaches the shutdown temperature. The 'warning-temperature' and 'shutdown-temperature' are maintained in the NVRAM for the Netra CP2060/

CP2080 board (for warning and shutdown temperature values, see *Table 6-4*). Also, the `show-sensor` command at OBP displays the readings of all the temperature sensors on the board.

When the CPU temperature reaches the set warning temperature limit, the following message is displayed at the `ok` prompt at regular intervals,

```
<<< WARNING!!! Upper Non-critical - going high >>>
The current threshold setting is: < >
The current temperature is      : < >
```

When the CPU temperature reaches the set shutdown temperature limit, the following message is displayed at the `ok` prompt at regular intervals,

```
<<< !!! ALERT!!! Upper Critical - going high >>>
     The current threshold setting is:  < >
     The current temperature is      : < >
```

The warning and shutdown temperature values provided are the OBP default values. A user can change these values by changing the corresponding NVRAM variable values and resetting the system hardware or software.

```
ok setenv warning-temperature  <new_value>
ok setenv shutdown-temperature <new_value>
ok reset-all
```

The `<new_value>` is a decimal value for a new temperature limit. The OBP then uses the new temperature limits after the system reset.

**Caution –** Be careful when setting the temperature parameters. Setting the `warning-temperature` and `shutdown-temperature` values too high will leave the system unprotected against overheating. Setting the temperature too low may cause the Netra CP2060/CP2080 board to send error messages continuously.

## 6.6.2 ENUM Monitoring

The ENUM monitoring feature notifies the system host board when a satellite board or IO card is freshly inserted or extracted. If the system host board is at the OBP `ok` prompt, the NVRAM variable `env-monitor` is set at enabled, and an IO card or a satellite board is plugged in while the system is already powered on then a notification is received on the system board display console. The display states that there has been a change in the configuration of the system.

### 6.6.3 PCI_RESET# Polling on the Satellite Board

The ASM also provides PCI-RESET# polling. ASM enables the checking on the status of the PCI_RESET# on the system board and having the satellite board respond accordingly. For example, a hotswap cPCI chassis containing a system host board and a few CP2060/CP2080 satellite boards, that are all at the `ok` prompt: If the PCI_RESET# is reasserted by the system board, then the assertion is polled by the SMC on the satellite boards. The satellite board then does an automatic `reset-all` on itself.

## 6.7 Determining Firmware Version

If the installed version is not current, update the OBP before continuing. The third character group (X) in OBP is the revision number.

### *From OBP*

To determine the installed OBP version, use the *.version* command at the `ok` prompt. An **example** of the screen print is shown below.

```
ok .version
SMC Firmware Release 3.9.7 TEST Platform ID 11
*** SMC FIRMWARE NOT FOR PRODUCTION ***
FPGA Version 1.0
PLD Version 4.1
Firmware CORE Release 1.0.9 created 2001/5/15 13:13
Release 4.0 Version 394 created 2001/05/21 09:45
cPOST version 1.0.13 created 2001/5/14
CORE 1.0.9 2001/05/15 13:13
```

### *If Running Solaris Software*

Use the *prtconf* command at the `<machine_name>` prompt.

```
<machine_name>% /usr/bin/prtconf -V
OBP 4.0.xx <creation date>
```

# 6.8 Field CORE / OBP Firmware Upgrade

This firmware can only be upgraded when operating at the OBP level, that is, at the ok prompt. The following procedure gives the steps to update firmware on the target system.

---

**Note –** See information for sequencing of SMC and board upgrade information provided on the product website at :
*http://www.sun.com/microelectronics/commprovider/cp2060/*

---

▼

1. **Download the latest Netra CP2060/CP2080 host firmware binaries**

2. **Bring the system down to OBP level**

   If your CP2060 / CP2080 is currently running Solaris software, become superuser and issue the command:

   ```
   $ uadmin 2 0
   ```

3. **Check the present firmware revision**

   Check the current firmware revision on the target system by typing:

   ```
   ok .version
   ```

   A *typical* output is:

   ```
   ok .version
   SMC Firmware Release 3.2.2 Platform ID 11
   FPGA Version 0.8
   PLD Version 2.4
   Firmware CORE Release 0.0.16 created 2000/12/1 18:19
   Release 4.0 Version 26 created 2000/11/29 18:39
   CORE 0.0.87 2000/12/01 18:19
   ok
   ```

4. **Disable autoboot; then reset**

   Disable autoboot and reset the system by means of the commands:

   ```
   ok setenv auto-boot? false
   ok reset-all
   ```

   The system automatically takes a reset.

5. **Flash update your firmware**

```
ok flash-update <device;> <obp-file-path>/<obp-latest-binary>
ok smc-flash-update <device;> <smc-file-path>/<smc-latest-binary>
```

The system should automatically take a reset. If it does not, power cycle it.

6. **Check the firmware revision**

   Check the firmware revision by typing:

   ```
   ok .version
   ```

   The form of the output appears as in Step 3. Ensure that the version information shows up as expected. If not, please attempt the OBP upgrade once more.

7. **Enable auto booting and reset the system**

   Enable auto booting by typing:

   ```
   ok setenv auto-boot? true
   ```

   and reset the system to boot Solaris software:

   ```
   ok reset-all
   ```

   Please contact your service personnel if you face any problems.

---

**Note –** Solaris scripts are also available to upgrade core OBP firmware.

---

## 6.9 SMC Firmware

The field upgradeable SMC firmware supports features such as CP2060/CP2080 board resources, temperature monitoring, control of the power module, IPMI communication with other boards, PCI reset modes of operation, hotswap capability and watchdog timer heartbeat mechanism. The SMC firmware also has its own built-in self test at power up. The SMC consists of DS80CH11, which is an 8051 compatible chip and the WS833, the memory chip. Inside WS833, there are the main flash and the boot flash and SRAM for data storage. The host CPU sends commands and data to SMC via the Ebus. For more details on the SMC subsystem please see Section 3.6, *System Management Controller*.

The SMC architecture allows the update of the SMC firmware. SMC firmware is only updated from the OBP. This feature is used to modify SMC firmware during a field upgrade, for fixing bugs, adding enhancements/new features, or providing special code for a specific OEM customer.

The SMC is capable of performing a flash update on both, main and boot flash. The main flash can flash update the boot flash, and the boot flash can flash update the main flash. The boot code contains the bare minimum code to be able to let the

system boot to ok prompt in the event of the main flash failure, and be able to switch from boot flash to main flash for execution. Therefore any attempt to perform flash update to the boot flash is considered risky and should not be done too often.

The CORE/OBP code has support for recovery in case of SMC flash update failure. When it detects that SMC is running from boot code, it automatically goes to the ok prompt, and the user can do a flash update. Any other commands sent to SMC will not be allowed at this time.

For full details on SMC firmware and detailed commands, please refer to the Netra CP2060/CP2080 website:

*http://www.sun.com/microelectronics/commprovider/cp2060/*

---

**Note –** Due to interdependency between OBP CORE, SMC and hardware, the user must take into account compatibility between various parts of the system among different version numbers when performing flash update. Please refer to the release note at the product website for the latest information and compatibility between firmware and hardware components.

---

## 6.9.1 SMC Firmware Reset Modes for System Slot and Peripheral Slot Operations

This section provides information on the various modes of reset available on the Netra CP2060/CP2080 board when used in the different roles and CompactPCI slots. *Table 6-7* describes the available modes of operation in response to a reset request on the CompactPCI backplane. Determination of system or peripheral/satellite operation is made from the state of the cPCI backplane SYSEN# signal as per the *PICMG 2.0 R3.0 Specification*. The RESET# signal affects only the PCI component of the cPCI bus. Please refer to the Netra CP2060/CP2080 website for detailed information on reset modes:

*http://www.sun.com/microelectronics/commprovider/cp2060/*

**Table 6-7** Reset Operating Modes

| Reset Mode | System Slot | Peripheral / Satellite Slot |
|---|---|---|
| 11 | Standard system slot operation -- the board generates normal RESET# and PCI signalling for the backplane in its role as system controller | Backplane reset is propagated to the UltraSPARC-IIe 21554 NTB and other reset table components on the board. This results in a complete reset of the UltraSPARC section of the board. |
| 22 | Standalone mode - The board asserts a constant RESET# but no PCI clocking for the cPCI bus, and does not respond to any PCI signalling on the backplane | Standalone mode - The local cPCI bridge is held in reset, isolated from the cPCI bus. the board does not respond to any PCI signalling on the cPCI bus. |
| 66[1] | Standard system slot operation -- the board generates normal RESET# and PCI signalling for the backplane in its role as system controller | Standalone mode - The local cPCI bridge is held in reset, isolated from the cPCI bus. The board does not respond to any PCI signalling on the cPCI bus. |

1. Reset Mode 66 is the default setting for the Netra CP2060 / CP2080 boards.

Users may reprogram the operating mode from the OBP prompt, then reboot (power cycle) the entire system in order for the new reset modes to take effect.

**Caution –** Some of these modes may be incompatible with various PICMG specifications, and customers may use these modes at their own risk.

## 6.9.2 SMC Configuration Block

The SMC power-on behavior and other attributes are stored in an 8 byte configuration block. This configuration block is stored in an accessible Serial $I^2C$ EEPROM (see *Figure 3-2* or *Figure 3-3*). In the absence of this configuration block, SMC boots up in a default mode. For this purpose, at the OBP level, we have provided two commands `set-smcenv` and `show-smcenv` in the SMC node. These commands are used to set or get the SMC configuration block.

### 6.9.2.1 Command Usage Detail

To change the settings on the configuration block, read the block using the `show-smcenv.` If not the desired setting, then use the `set-SMCenv` to change the SEEPROM configuration block. Some examples are given below:

To change the grant/ignore request byte, check the current setting at the OBP ok prompt and follow the steps: (Look for byte xyz)

```
ok show-smcenv                              ; Returns 8 bytes + completion code
ok a b c d xyz e f g 0                       ; This is the block all in HEX
x ==> bits 0:2  001 = no confirmation          ; 010 = send confirmation
bit 3 : reserved
y ==> bits 4:5   00 = Grant XIR                ;  01 = Ignore XIR
z ==> bits 6:7   00 = Grant Push button reset ;  01 = Ignore Push button reset
```

Example:

```
To configure the setting to grant XIR and ignore PB and no confirmation for
SIR:
x = 001 no confirmation for SIR
y = 00  grant XIR
z = 01  ignore PB
ok g f e 41 d c b a set-SMCenv       ; Change the configuration block
ok 0                                    ; completed  *** Now POWER CYCLE ***
```

---

**Note –** Each the setting on the configuration block can change the behavior of the board significantly. (Refer to the host-SMC specification).

---

During OBP start up sequence, and before the PCI probe, OBP checks for a valid SMC configuration block. If it does not find a valid configuration block (i.e configuration version is not equal to 1), then OBP instructs the SMC to program the configuration block with the following default settings,

```
Config version:   1
Backplane info:   0
Reset mode:       66
SIR & XIR:        2
Health control:   0
Health status:    0
byte 7:           0
byte 8:           0
```

# 6.10 Firmware Diagnostics

The firmware contains a comprehensive set of hardware diagnostic modules that provide tests for most situations. Chapter 6, *Firmware*, shows the control-flow relationship of the diagnostic modules with the system firmware. The SunValidation Test Suite (SunVTS) package can be executed from within the Solaris software if more tests are required. For more information, see: Chapter 7, *System Diagnostics—SunVTS*.

The firmware diagnostic modules are:

- Basic POST (BPOST)
- Comprehensive POST (CPOST) -- optional
- Extended POST (EPOST) -- OEM supplied dropin
- OBDiag

The firmware diagnostics cover address and data bits on all system buses and exercise the function of the major hardware resources on the board.

Diagnostics can be performed at OBP level by using the *obdiag* command, or by typing individual test commands at the *ok* prompt. These test suites are similar to those in earlier OBP versions but they are comprised of dropins that can be placed by the user. See the reference to the *OpenBoot 3.x Command Reference Manual* listed in *Bibliography*.

## 6.10.1 Setting Diagnostic Levels

The user interface in terms of running POST at minimum or maximum remains the same. BPOST is embedded within Firmware CORE and is executed when the OBP environment variable, *diag- switch* is set to *true* and *diag- level* set to *min*. Similarly CPOST (and EPOST if it is present) is executed when diag-level is set to *max*. The permutations are shown in *Table 6-5*.

CPOST, and Extended POST are clients of Firmware CORE.

## 6.10.2 Basic POST (BPOST)

BPOST is integrated into Firmware CORE. It can provide on-demand diagnostic services in response to IPMI requests from the System Management Bus.

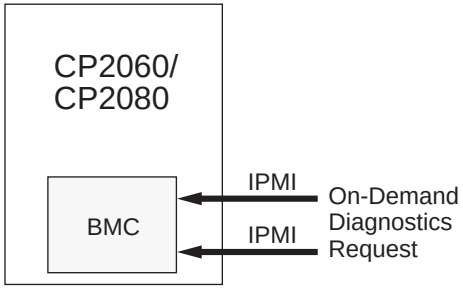



**Figure 6-3** Basic POST Services

BPOST consists of two parts:

The first part of BPOST executes from flash memory. It is designed to validate enough of the system resources to be able to run Firmware CORE in main memory (System RAM). If this test phase is passed, BPOST is also copied into system RAM. BPOST runs when the `diag-switch?` is set to `true` (see *Table 6-5*).

The part of BPOST executed from flash includes basic tests for the items:

- NVRAM
- I-cache and D-cache
- MMU
- FPU
- L2-cache tag and RAM
- Data lines
- CORE memory

The second part is performed after Firmware CORE is copied into main RAM. This part of BASIC POST executed from RAM includes:

- Memory address line test; this test assumes that the CPU, MMU, and FPU are functional.
- ECC block memory test; verifies main memory with block write and ECC checking. This test assumes that the CPU, MMU, and FPU are functional.

## 6.10.3 Comprehensive POST (CPOST)

Comprehensive POST (CPOST) is a client of Firmware CORE. It is a dropin module invoked by Firmware CORE and contains enhanced diagnostics for the CPU and on-board devices.

The execution of CPOST is optional and can be selectively controlled by an environment variable—see *Table 6-5*. CPOST runs after BPOST. To run CPOST, set the environment variables diag-switch? to *true* and diag-level set to *max*

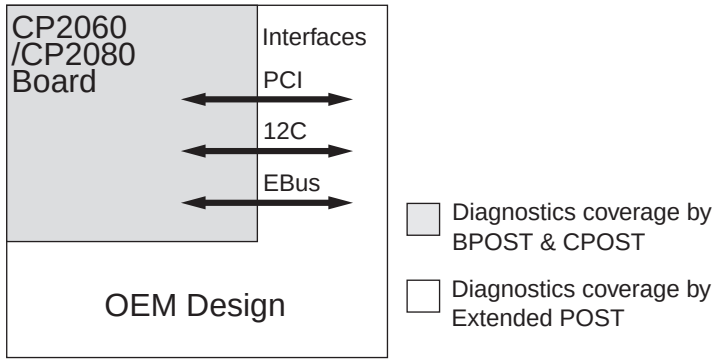
CPOST tests comprise:

- Memory stress test; advanced main memory test
- Basic PBM, IOMMU test
- Basic Advanced PCI Bridge APB test
- PCI/E-bus/Ethernet/SuperIO tests
- System Management Controller test

Execution passes to EPOST (if it exists) or undergoes a software reset which sends it back to Firmware CORE. From this point, execution enters OBP (since diagnostics are only executed at power on reset).

## 6.10.4 Extended POST

Extended POST enables OEMs to provide additional firmware diagnostics for their hardware within a Netra CP2060/CP2080-based system. Extended POST is a dropin module invoked by CPOST and is also a client of Firmware CORE from which it uses trap based services.




Extended POST enables OEMs to add diagnostic support for their H/W in a CP2060/CP2080 system.

**Figure 6-4** POST Enables OEMs to add Diagnostics

The conditions for execution of EPOST are:

- after CPOST completes, if there were any error, execution returns to the OBP with its standard interface. If BPOST tests are passed and `diag-level` is not set to *max*, execution passes to OBP.
- If the `diag-level` is *max*, the CPOST code checks if there is an EPOST dropin in flash or user PROM area. If it does not find one, it displays a message:

  ```
  There is no extended POST in this system
  ```

- If CPOST finds an EPOST dropin, it loads it into memory and runs it.

Before passing control to EPOST, CPOST creates a list of pointers of vital functions and passes these to EPOST.

## 6.10.5 OpenBoot PROM On-Board Diagnostics

The OBP on-board diagnostics reside in the OBP dropin. These diagnostics are described fully in the *OpenBoot 3.x Command Reference Manual*—see *Bibliography*.

To execute the OBP on-board diagnostics, the system must be at the ok prompt. The OBP on-board diagnostics comprise:

- watch-clock
- watch-net and watch-net-all
- probe-scsi
- test <device path>
- test-all

## 6.10.6 OpenBoot Diagnostics (OBDiag)

The OpenBoot Diagnostics (OBDiag) are an enhancement of the traditional system tests. They reside in Forth script in a dropin and are invoked with an interactive tool that is started from the ok prompt by typing `obdiag`.

When OBDiag is started, the OBDiag menu shown in *Table 6-8* is displayed.

**Table 6-8** OBDiag -- Diagnostics Test printout

| obdiag | | |
|---|---|---|
| 1 ebus@1 | 2 ebus@3 | 3 flashprom@10,0 |
| 4 flashprom@10,400000 | 5 network@1,1 | 6 network@3,1 |
| 7 usb@1,3 | 8 usb@3,3 | |
| Commands: test test-all except help what printenvs setenv versions exit | | |

When at the `obdiag` prompt, typing `test-all` would display a printout similar to the following:

```
obdiag> test-all
Hit the spacebar to interrupt testing
Testing /pci@1f,0/pci@1,1/ebus@1 ..................................... passed
Testing /pci@1f,0/pci@1,1/ebus@3 ..................................... passed
Testing /pci@1f,0/pci@1,1/ebus@1/flashprom@10,0 ...................... passed
Testing /pci@1f,0/pci@1,1/ebus@1/flashprom@10,400000 ................. passed
Testing /pci@1f,0/pci@1,1/network@1,1 ................................ passed
Testing /pci@1f,0/pci@1,1/network@3,1 ................................ passed
Testing /pci@1f,0/pci@1,1/usb@1,3 .................................... passed
Testing /pci@1f,0/pci@1,1/usb@3,3
```

CHAPTER 7

# System Diagnostics—SunVTS

Sun Validation Test Suite (SunVTS™) is a comprehensive software package that tests and validates the Netra CP2060/CP2080 by verifying the configuration and function of most hardware controllers and devices on the board. SunVTS is used to validate a system during development, production, inspection, troubleshooting, periodic maintenance and system or subsystem stressing. SunVTS can be tailored to run on various types of machines ranging from desktops to servers with modifiable test instances and processor affinity features.

You can perform high-level system testing by using the appropriate version of SunVTS. For detailed information on SunVTS support, refer to the Netra CP2060/CP2080 website:

*http://www.sun.com/microelectronics/commprovider/cp2060/*

## 7.1 Distribution of SunVTS

This section provides some information on distribution of SunVTS.

- SunVTS may be downloaded by following Netra CP2060/CP2080 links from the following URL:

*http://www.sun.com/microelectronics/commprovider/cp2060/*

You can also refer to the appropriate SunVTS manual (see Bibliography).

- Ensure the SunVTS version is compatible with the Solaris Operating Environment version being used.

  Solaris™ 8 Update 3 Operating Environment is compatible with SunVTS version 4.2

- Information on the version of SunVTS installed can be found in the file: `/opt/SUNWvts/bin/.version`
- These packages are located on the Sun Computer Systems Supplement CD-ROM that ships with the Solaris Operating Environment release. Documentation links on the SunVTS products and details on additional SunVTS tests specific to Netra CP2060/CP2080 boards can be found at the following website:

*http://www.sun.com/microelectronics/commprovider/cp2060/*

To obtain SunVTS documentation, contact your local customer service representative or field applications engineer (FAE).

## 7.2 Installing and Starting SunVTS

For security reasons only a superuser is permitted to run SunVTS. Installation and starting instructions are included with the software when it is downloaded.

CHAPTER 8

# Connectors, Pinouts and Switch Settings

This chapter provides information on Netra CP2060/CP2080 connectors, switch settings and pinouts.

## 8.1 PMC Connector

*Figure 8-1*and *Figure 8-2* illustrates PMC port connectors. The tables show contact allocations.

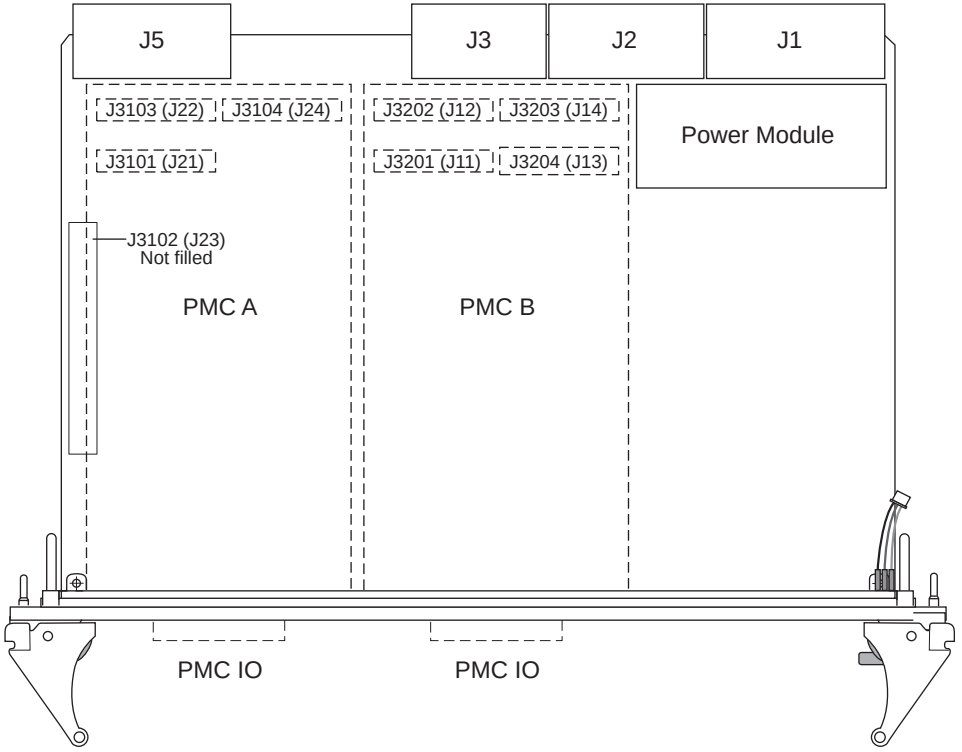



**Figure 8-1** Netra CP2060 PMC Port Connectors

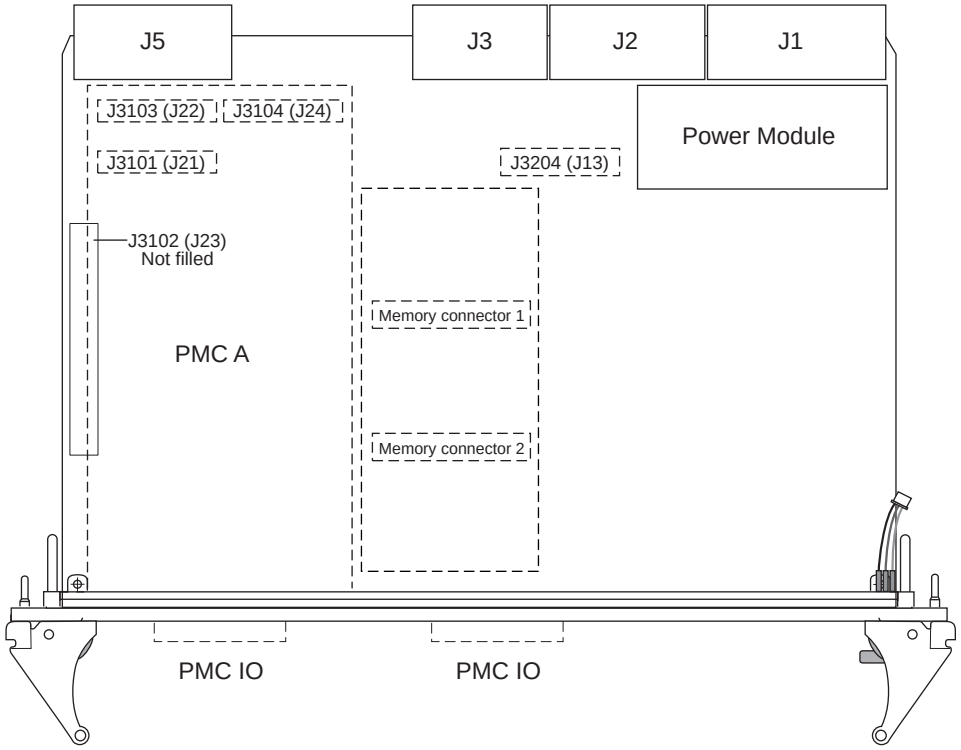



**Figure 8-2** Netra CP2080 PMC Port Connectors

.

**Table 8-1** PMC-A J21 Connector Interface

| Pin | Description | Pin | Description |
|---|---|---|---|
| 1 | TCK; JTAG clock signal. | 2 | -12V |
| 3 | GND | 4 | PMC_A_INT_A_L |
| 5 | PMC_A_INT_B_L | 6 | PMC_A_INT_C_L |
| 7 | PMC_BUSMODE1_L[1] | 8 | VCC (5V) |
| 9 | PMC_A_INT_D_L | 10 | NC |
| 11 | GND | 12 | NC |
| 13 | PMC_CLK | 14 | GND |
| 15 | GND | 16 | PMC_GNT_L |
| 17 | PMC_REQ_L | 18 | VCC |
| 19 | LOCAL_VIO | 20 | PCI_B_AD<31> |

**Table 8-1** PMC-A J21 Connector Interface

| Pin | Description | Pin | Description |
|---|---|---|---|
| 21 | PCI_B_AD<28> | 22 | PCI_B_AD<27> |
| 23 | PCI_B_AD<25> | 24 | GND |
| 25 | GND | 26 | PCI_B_CBE3_L |
| 27 | PCI_B_AD<22> | 28 | PCI_B_AD<21> |
| 29 | PCI_B_AD<19> | 30 | VCC |
| 31 | LOCAL_VIO | 32 | PCI_B_AD<17> |
| 33 | PCI_B_FRAME_L | 34 | GND |
| 35 | GND | 36 | PCI_B_IRDY_L |
| 37 | PCI_B_DEVSEL_L | 38 | VCC |
| 39 | GND | 40 | PCI_B_LOCK_L |
| 41 | PMC_SDONE | 42 | PMC_SB0_L |
| 43 | PCI_B_PAR | 44 | GND |
| 45 | LOCAL_VIO | 46 | PCI_B_AD<15> |
| 47 | PCI_B_AD<12> | 48 | PCI_B_AD<11> |
| 49 | PCI_B_AD<9> | 50 | VCC |
| 51 | GND | 52 | PCI_B_CBE_L<0> |
| 53 | PCI_B_AD<6> | 54 | PCI_B_AD<5> |
| 55 | PCI_B_AD<4> | 56 | GND |
| 57 | LOCAL_VIO | 58 | PCI_B_AD<3> |
| 59 | PCI_B_AD<2> | 60 | PCI_B_AD<1> |
| 61 | PCI_B_AD<0> | 62 | VCC |
| 63 | GND | 64 | PCI_B_REQ64_L |

1. BUSMODE signals require a pull-up

**Table 8-2** PMC-A J22 Connector Interface

| Pin | Description | Pin | Description |
|---|---|---|---|
| 1 | +12V | 2 | JTAG_PMC_RST_L |
| 3 | TMS | 4 | PMC_TDO |
| 5 | PMC_TDI | 6 | GND |
| 7 | GND | 8 | NC |
| 9 | NC | 10 | NC |

**Table 8-2** PMC-A J22 Connector Interface

| Pin | Description | Pin | Description |
|---|---|---|---|
| 11 | PMC_BUSMODE2_L | 12 | VDD (3.3V) |
| 13 | PCI_B_RST_L | 14 | PMC_BUSMODE3_L |
| 15 | VDD | 16 | PMC_BUSMODE4_L |
| 17 | NC | 18 | GND |
| 19 | PCI_B_AD<30> | 20 | PCI_B_AD<29> |
| 21 | GND | 22 | PCI_B_AD<26> |
| 23 | PCI_B_AD<24> | 24 | VDD |
| 25 | PCI_B_IDSEL | 26 | PCI_B_AD<23> |
| 27 | VDD | 28 | PCI_B_AD<20> |
| 29 | PCI_B_AD<18> | 30 | GND |
| 31 | PCI_B_AD<16> | 32 | PCI_B_CBE_L<2> |
| 33 | GND | 34 | NC |
| 35 | PCI_B_TRDY_L | 36 | VDD |
| 37 | GND | 38 | PCI_B_STOP_L |
| 39 | PCI_B_PERR_L | 40 | GND |
| 41 | VDD | 42 | PCI_B_SERR_L |
| 43 | PCI_B_CBE_L<1> | 44 | GND |
| 45 | PCI_B_AD<14> | 46 | PCI_B_AD<13> |
| 47 | GND | 48 | PCI_B_AD<10> |
| 49 | PCI_B_AD<8> | 50 | VDD |
| 51 | PCI_B_AD<7> | 52 | NC |
| 53 | VDD | 54 | NC |
| 55 | NC | 56 | GND |
| 57 | NC | 58 | NC |
| 59 | GND | 60 | NC |
| 61 | PCI_B_ACK64_L | 62 | VDD |
| 63 | GND | 64 | NC |

**Table 8-3** PMC-A J23 Connector Pin Assignments (J23)

| Pin | Signal | Signal | Pin |
|---|---|---|---|
| P1386.1 standard reserves Jn3 64-pin connector for PCI 64-bit extensions. It is not fitted on these boards. | | | |

**Table 8-4** PMC-A J24 Connector Interface

| Pin | Description |
|---|---|
| 1-64 | PMC_A_IO<1-64> are user defined IO pins |

## 8.1.1 PMC-B Interface

The PMC-B interface is comprised of four connectors. They are connectors J11, J12, and J14 which conform to the Common Mezzanine Card (CMC) specification and J13 which is proprietary I/O connections.

**Table 8-5** PMC-B J11 Connector Interface on Netra CP2060 ONLY

| Pin | Description | Pin | Description |
|---|---|---|---|
| 1 | TCK; JTAG clock signal. | 2 | -12V |
| 3 | GND | 4 | PMC_B_INT_A_L |
| 5 | PMC_B_INT_B_L | 6 | PMC_B_INT_C_L |
| 7 | PMC_BUSMODE1_L | 8 | VCC (5V) |
| 9 | PMC_B_INT_D_L | 10 | NC |
| 11 | GND | 12 | NC |
| 13 | PMC_CLK | 14 | GND |
| 15 | GND | 16 | PMC_GNT_L |
| 17 | PMC_REQ_L | 18 | VCC |
| 19 | LOCAL_VIO | 20 | PCI_B_AD<31> |
| 21 | PCI_B_AD<28> | 22 | PCI_B_AD<27> |
| 23 | PCI_B_AD<25> | 24 | GND |
| 25 | GND | 26 | PCI_B_CBE3_L |
| 27 | PCI_B_AD<22> | 28 | PCI_B_AD<21> |
| 29 | PCI_B_AD<19> | 30 | VCC |
| 31 | LOCAL_VIO | 32 | PCI_B_AD<17> |

**Table 8-5** PMC-B J11 Connector Interface on Netra CP2060 ONLY

| Pin | Description | Pin | Description |
|---|---|---|---|
| 33 | PCI_B_FRAME_L | 34 | GND |
| 35 | GND | 36 | PCI_B_IRDY_L |
| 37 | PCI_B_DEVSEL_L | 38 | VCC |
| 39 | GND | 40 | PCI_B_LOCK_L |
| 41 | PMC_SDONE | 42 | PMC_SB0_L |
| 43 | PCI_B_PAR | 44 | GND |
| 45 | LOCAL_VIO | 46 | PCI_B_AD<15> |
| 47 | PCI_B_AD<12> | 48 | PCI_B_AD<11> |
| 49 | PCI_B_AD<9> | 50 | VCC |
| 51 | GND | 52 | PCI_B_CBE_L<0> |
| 53 | PCI_B_AD<6> | 54 | PCI_B_AD<5> |
| 55 | PCI_B_AD<4> | 56 | GND |
| 57 | LOCAL_VIO | 58 | PCI_B_AD<3> |
| 59 | PCI_B_AD<2> | 60 | PCI_B_AD<1> |
| 61 | PCI_B_AD<0> | 62 | VCC |
| 63 | GND | 64 | PCI_B_REQ64_L |

**Table 8-6** PMC-B J12 Connector Interface on Netra CP2060 ONLY

| Description | Pin | Pin | Description |
|---|---|---|---|
| +12V | 1 | 2 | JTAG_PMC_RST_L |
| TMS | 3 | 4 | PMC_TDO |
| PMC_TDI | 5 | 6 | GND |
| GND | 7 | 8 | NC |
| NC | 9 | 10 | NC |
| PMC_BUSMODE2_L | 11 | 12 | VDD (3.3V) |
| PCI_B_RST_L | 13 | 14 | PMC_BUSMODE3_L |
| VDD | 15 | 16 | PMC_BUSMODE4_L |
| NC | 17 | 18 | GND |
| PCI_B_AD<30> | 19 | 20 | PCI_B_AD<29> |
| GND | 21 | 22 | PCI_B_AD<26> |
| PCI_B_AD<24> | 23 | 24 | VDD |

**Table 8-6** PMC-B J12 Connector Interface on Netra CP2060 ONLY

| Description | Pin | Pin | Description |
|---|---|---|---|
| PCI_B_IDSEL | 25 | 26 | PCI_B_AD<23> |
| VDD | 27 | 28 | PCI_B_AD<20> |
| PCI_B_AD<18> | 29 | 30 | GND |
| PCI_B_AD<16> | 31 | 32 | PCI_B_CBE_L<2> |
| GND | 33 | 34 | NC |
| PCI_B_TRDY_L | 35 | 36 | VDD |
| GND | 37 | 38 | PCI_B_STOP_L |
| PCI_B_PERR_L | 39 | 40 | GND |
| VDD | 41 | 42 | PCI_B_SERR_L |
| PCI_B_CBE_L<1> | 43 | 44 | GND |
| PCI_B_AD<14> | 45 | 46 | PCI_B_AD<13> |
| GND | 47 | 48 | PCI_B_AD<10> |
| PCI_B_AD<8> | 49 | 50 | VDD |
| PCI_B_AD<7> | 51 | 52 | NC |
| VDD | 53 | 54 | NC |
| NC | 55 | 56 | GND |
| NC | 57 | 58 | NC |
| GND | 59 | 60 | NC |
| PCI_B_ACK64_L | 61 | 62 | VDD |
| GND | 63 | 64 | NC |

**Table 8-7** PMC-B J13 Connector Pin Assignments on Netra CP2080 ONLY

P1386.1 standard reserves the J13 80-pin connector for PCI 64-bit extensions. The connector pin descriptions are proprietary information.

On the Netra CP2060, the J13 connector is not present.

**Table 8-8** PMC-B J14 Connector Interface on Netra CP2060 ONLY

| Pin | Description |
|---|---|
| 1-64 | PMC_B_IO<1-64> are user defined IO pins |

# 8.2 CompactPCI Backplane Connectors

*Figure 8-3* shows contact numbering as seen from the back of the Netra CP2060/CP2080 board.

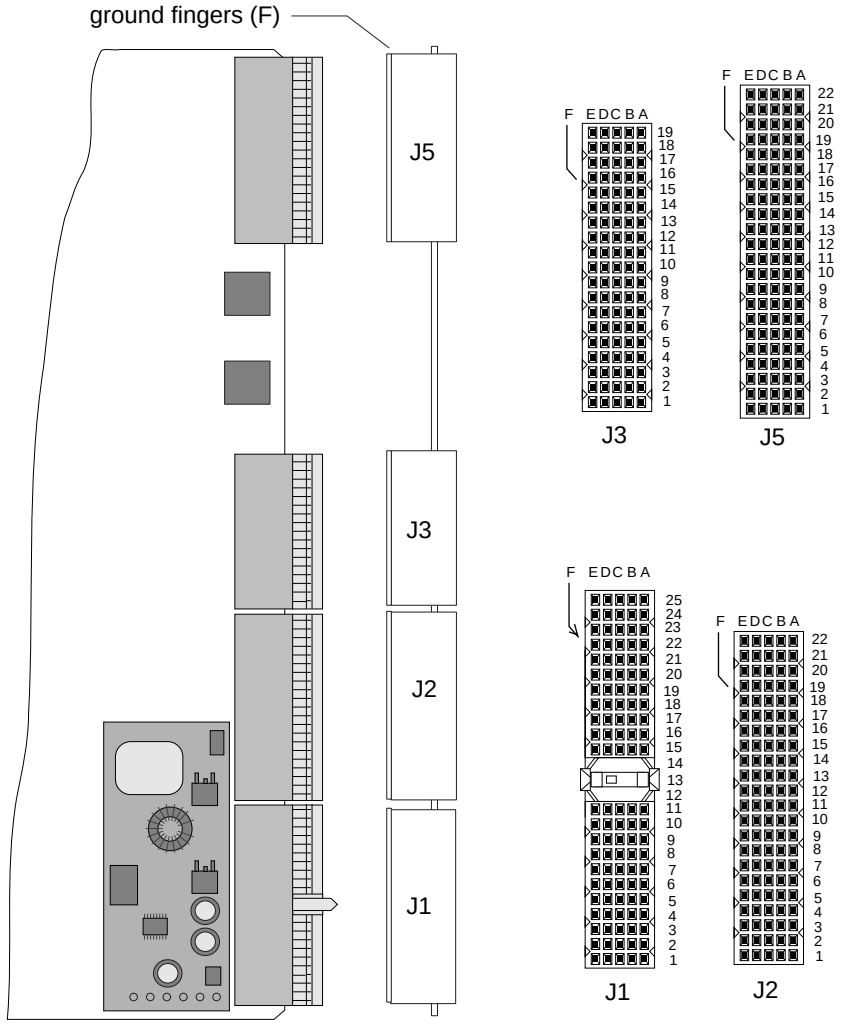



**Figure 8-3** CompactPCI Host Board Connector Contact Numbering

---

**Note –** A blue key on the J1 connector indicates a 5V PCI.

---

## 8.2.1 CompactPCI Connectors

The J4 connector is not populated on the Netra CP2060/CP2080 boards.

**Table 8-9** J1/P1 Connector Pin Assignments

| Pin | Row Z | Row A | Row B | Row C | Row D | Row E | Row F |
|---|---|---|---|---|---|---|---|
| 25 | GND | +5V | REQ64# | ENUM# | +3.3V | +5V | GND |
| 24 | GND | AD[1] | +5V | +5V | AD[0] | ACK64# | GND |
| 23 | GND | +3.3V | AD[4] | AD[3] | +5V | AD[2] | GND |
| 22 | GND | AD[7] | GND | +3.3V | AD[6] | AD[5] | GND |
| 21 | GND | +3.3V | AD[9] | AD[8] | M66EN | C/BE[0]# | GND |
| 20 | GND | AD[12] | GND | +5V | AD[11] | AD[10] | GND |
| 19 | GND | +3.3V | AD[15] | AD[14] | GND | AD[13] | GND |
| 18 | GND | SERR# | GND | +3.3V | PAR | C/BE[1]# | GND |
| 17 | GND | +3.3V | IPMB SCL | IPMB SDA | GND | PERR# | GND |
| 16 | GND | DEVSEL# | GND | +5V | STOP# | LOCK# | GND |
| 15 | GND | +3.3V | FRAME# | IRDY# | BD_SEL# | TRDY# | GND |
| 14 | Key | | | | | | Key |
| 13 | Key | | | | | | Key |
| 12 | Key | | | | | | Key |
| 11 | GND | AD[18] | AD[17] | AD[16] | GND | C/BE[2]# | GND |
| 10 | GND | AD[21] | GND | +3.3V | AD[20] | AD[19] | GND |
| 9 | GND | C/BE[3]# | IDSEL | AD[23] | GND | AD[22] | GND |
| 8 | GND | AD[26] | GND | +5V | AD[25] | AD[24] | GND |
| 7 | GND | AD[30] | AD[29] | AD[28] | GND | AD[27] | GND |
| 6 | GND | REQ0# | GND | +3.3V | CLK0 | AD[31] | GND |
| 5 | GND | | | RST# | GND | GNT0# | GND |
| 4 | GND | IPMB_PWR | GND | +5V | | | GND |
| 3 | GND | INTA# | INTB# | INTC# | +5V | INTD# | GND |
| 2 | GND | | +5V | | | | GND |
| 1 | GND | +5V | -12V | | +12V | +5V | GND |

**Note –** Gray fill indicates backplane long pin. Thick box border indicates short pin. All other pins are medium pins.

**Table 8-10** J2/P2 Connector Pin Assignments

| Pin | Row Z | Row A | Row B | Row C | Row D | Row E | Row F |
|---|---|---|---|---|---|---|---|
| 22 | GND | GA4 | GA3 | GA2 | GA1 | GA0 | GND |
| 21 | GND | CLK6 | GND | BD_TYPE 0 | BD_TYPE 1 | BD_OFF_ OUT | GND |
| 20 | GND | CLK5 | GND | | GND | BD_OFF_ IN | GND |
| 19 | GND | GND | GND | I2C_SDA | I2C_SCK | | GND |
| 18 | GND | | | | GND | | GND |
| 17 | GND | | GND | PRST# | REQ6# | GNT6# | GND |
| 16 | GND | | | DEG# | GND | | GND |
| 15 | GND | | GND | FAL# | REQ5# | GNT5# | GND |
| 14 | GND | AD[35] | AD[34] | AD[33] | GND | AD[32] | GND |
| 13 | GND | AD[38] | GND | +5Va | AD[37] | AD[36] | GND |
| 12 | GND | AD[42] | AD[41] | AD[40] | GND | AD[39] | GND |
| 11 | GND | AD[45] | GND | +5Va | AD[44] | AD[43] | GND |
| 10 | GND | AD[49] | AD[48] | AD[47] | GND | AD[46] | GND |
| 9 | GND | AD[52] | GND | +5Va | AD[51] | AD[50] | GND |
| 8 | GND | AD[56] | AD[55] | AD[54] | GND | AD[53] | GND |
| 7 | GND | AD[59] | GND | +5Va | AD[58] | AD[57] | GND |
| 6 | GND | AD[63] | AD[62] | AD[61] | GND | AD[60] | GND |
| 5 | GND | C/BE[5]# | GND | +5Va | C/BE[4]# | PAR64 | GND |
| 4 | GND | +5V[a] | | C/BE[7]# | GND | C/BE[6]# | GND |
| 3 | GND | CLK4 | GND | GNT3# | REQ4# | GNT4# | GND |
| 2 | GND | CLK2 | CLK3 | SYSEN# | GNT2# | REQ3# | GND |
| 1 | GND | CLK1 | GND | REQ1# | GNT1# | REQ2# | GND |

**Table 8-11** J3/P3 Connector Pin Assignments

| Pin | Row Z | Row A | Row B | Row C | Row D | Row E | Row F |
|---|---|---|---|---|---|---|---|
| 19 | GND | PMCB-1 | PMCB-2 | PMCB-3 | PMCB-4 | PMCB-5 | GND |
| 18 | GND | PMCB-6 | PMCB-7 | PMCB-8 | PMCB-9 | PMCB-10 | GND |
| 17 | GND | PMCB-11 | PMCB-12 | PMCB-13 | PMCB-14 | PMCB-15 | GND |
| 16 | GND | PMCB-16 | PMCB-17 | PMCB-18 | PMCB-19 | PMCB-20 | GND |
| 15 | GND | PMCB-21 | PMCB-22 | PMCB-23 | PMCB-24 | PMCB-25 | GND |
| 14 | GND | PMCB-26 | PMCB-27 | PMCB-28 | PMCB-29 | PMCB-30 | GND |
| 13 | GND | PMCB-31 | PMCB-32 | PMCB-33 | PMCB-34 | PMCB-35 | GND |
| 12 | GND | PMCB-36 | PMCB-37 | PMCB-38 | PMCB-39 | PMCB-40 | GND |
| 11 | GND | PMCB-41 | PMCB-42 | PMCB-43 | PMCB-44 | PMCB-45 | GND |
| 10 | GND | PMCB-46 | PMCB-47 | PMCB-48 | PMCB-49 | PMCB-50 | GND |
| 9 | GND | PMCB-51 | PMCB-52 | PMCB-53 | PMCB-54 | PMCB-55 | GND |
| 8 | GND | PMCB-56 | PMCB-57 | PMCB-58 | PMCB-59 | PMCB-60 | GND |
| 7 | GND | PMCB-61 | PMCB-62 | PMCB-63 | PMCB-64 | V I/O | GND |
| 6 | GND | | | | GPIO1 | GPIO2 | GND |
| 5 | GND | | | | BKRST# OUT | BKRST# IN | GND |
| 4 | GND | | | | SMC RX | SMC TX | GND |
| 3 | GND | | | VISA RST# | PCI-B RST# | | GND |
| 2 | GND | | | | | | GND |
| 1 | GND | | | BP_XIR_L | | | GND |

**Note –** The J4 connector is not populated on the Netra CP2060/CP2080 boards.

**Table 8-12** J5/P5 Connector Pin Assignments

| Pin | Row Z | Row A | Row B | Row C | Row D | Row E | Row F |
|---|---|---|---|---|---|---|---|
| 22 | GND | PMCA-5 | PMCA-4 | PMCA-3 | PMCA-2 | PMCA-1 | GND |
| 21 | GND | PMCA-10 | PMCA-9 | PMCA-8 | PMCA-7 | PMCA-6 | GND |
| 20 | GND | PMCA-15 | PMCA-14 | PMCA-13 | PMCA-12 | PMCA-11 | GND |
| 19 | GND | PMCA-20 | PMCA-19 | PMCA-18 | PMCA-17 | PMCA-16 | GND |
| 18 | GND | PMCA-25 | PMCA-24 | PMCA-23 | PMCA-22 | PMCA-21 | GND |
| 17 | GND | PMCA-30 | PMCA-29 | PMCA-28 | PMCA-27 | PMCA-26 | GND |
| 16 | GND | PMCA-35 | PMCA-34 | PMCA-33 | PMCA-32 | PMCA-31 | GND |
| 15 | GND | PMCA-40 | PMCA-39 | PMCA-38 | PMCA-37 | PMCA-36 | GND |
| 14 | GND | PMCA-45 | PMCA-44 | PMCA-43 | PMCA-42 | PMCA-41 | GND |
| 13 | GND | PMCA-50 | PMCA-49 | PMCA-48 | PMCA-47 | PMCA-46 | GND |
| 12 | GND | PMCA-55 | PMCA-54 | PMCA-53 | PMCA-52 | PMCA-51 | GND |
| 11 | GND | PMCA-60 | PMCA-59 | PMCA-58 | PMCA-57 | PMCA-56 | GND |
| 10 | GND |  | PMCA-64 | PMCA-63 | PMCA-62 | PMCA-61 | GND |
| 9 | GND | RTS A | DTR A | RI A | GND | CTS A | GND |
| 8 | GND | DCD A | TXD A | RXD A | DSR A | +5V | GND |
| 7 | GND | RTS B | DTR B | RI B | DSR B | CTS B | GND |
| 6 | GND | DCD B | TXD B | RXD B | GND | GND | GND |
| 5 | GND | I2C_SCL |  | -12V | ENET1 RX (+) | ENET1 RX (-) | GND |
| 4 | GND | I2C_SDA | I2C_PWR | GND | ENET1 TX (+) | ENET1 TX (-) | GND |
| 3 | GND | USB2 (+) | USB2 (-) | GND | GND | GND | GND |
| 2 | GND | USB1 (+) | USB1 (-) | GND | ENET0 RX (+) | ENET0 RX (-) | GND |
| 1 | GND | GND | +3.3V | +12V | ENET0 TX (+) | ENET0 TX (-) | GND |

# 8.3 Switch Settings

A set of switches SW2501 and SW4101 are located between the heatsink and the front panel (see *Figure 8-4* for switch locations and direction of the arrows).

**Table 8-13** SW2501 Switch Settings

| Switch # | Setting | Description | Default |
|---|---|---|---|
| 1 | switch closed (in direction of arrow) | When switch closed, boot flash is programmable | yes |
| | switch open | When switch open, boot flash is not programmable | |
| 2 | switch closed (in direction of arrow) | When switch closed, the user flash is detected during OBP boot and the user flash is write enabled | yes |
| | switch open | When switch open, the user flash is not detected during OBP boot and the write-protect switch for user flash is enabled. | -- |

**Table 8-14** SW4101 Switch Settings

| Switch # | Setting | Description | Default |
|---|---|---|---|
| 1 | switch closed (in direction of arrow) | Always boot from (main) boot flash | yes |
| | switch open | Check SMC configuration block setting Byte 7 bits <3:2>[1] | -- |
| 2 | Reserved | | |

1. Check *Table 8-15* for details

**Table 8-15** SMC Configuration Block Setting Options; Byte 7 bits <3:2>

| Boot Device Setting | | | |
|---|---|---|---|
| **User Flash** | **ROMBO** | **User Flash** | **Main Flash** |
| 00 | 01 | 10 | 11 |
| 0..16MB=UF | 0..4MB=RB | 0..8MB=UF | 0..4MB=BF |
| | 4..12MB=UF | 8:12MB=BF | 4..12MB=UF |
| | 12..16MB=BF | 12..16MB=RB | 12..16MB=RB |

For details on SMC Configuration Block Setting see Section 6.9.2, *SMC Configuration Block*.

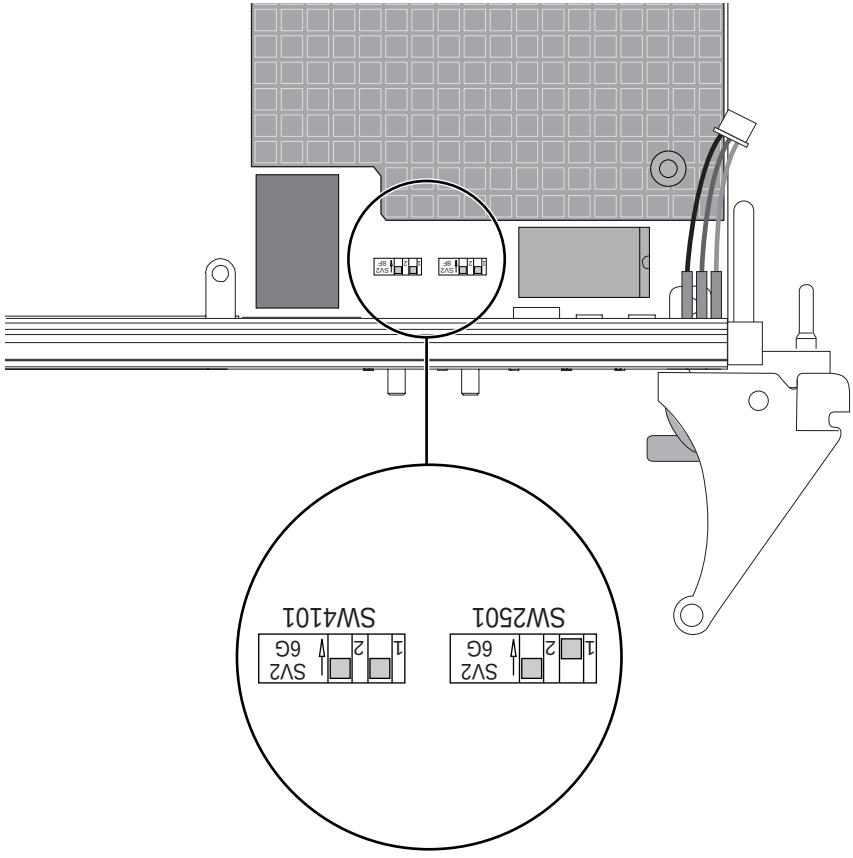



**Figure 8-4** Switches SW2501 and SW4101 on the Netra CP2060/CP2080 Board

CHAPTER 9

# Mechanical and Thermal Characteristics

This chapter provides detailed mechanical drawings and thermal characteristics for the Netra CP2060/CP2080 boards.

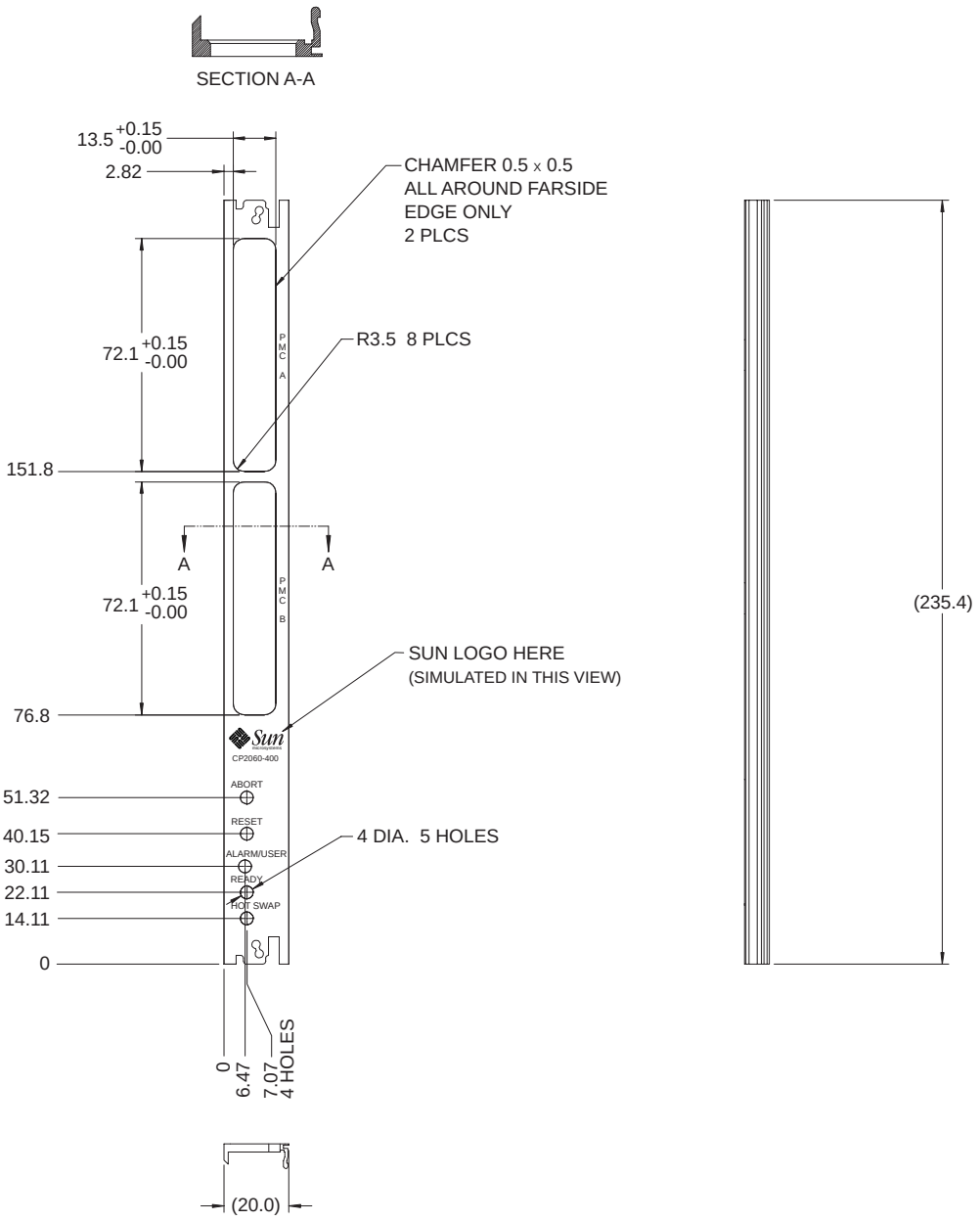



**Figure 9-1** Mechanical Illustration of the Netra CP2060 / CP2080 Front Panel

# 9.1 Thermal Characteristics

This section describes the general thermal guidelines for the CP2060/CP2080 boards in a typical configuration with an off-the-shelf cPCI chassis. The CP2060/CP2080 boards are designed to operate under certain environmental conditions. These conditions must be met in order for the board to function properly.

---

**Note –** Thermal measurements should only be performed by a qualified technician or a field service engineer.

---

**Table 9-1** Thermal Requirements for the CP2060/CP2080 500MHz

| Parameter | Temperature Range and Airflow | | Description |
|---|---|---|---|
| | CP2060/500MHz | CP2080/500MHz | |
| Max. heatsink temperature[1] | 78°C | 78°C | Temperature is measured by supergluing a thermocouple to the base (groove) of the heatsink between the fins, directly above the center of the microprocessor. |
| Heatsink performance[2] | 500MHz delta < 16.6°C (maximum) | 500MHz delta < 16.6°C (maximum) | Maximum delta is the heatsink temperature minus the ambient temperature to ensure proper cooling. |
| Airflow to meet NEBS Level 3 in OEM equipment[3] | 250 LFM | 250 LFM | Airflow greater than 250 LFM is needed if the airflow is not ducted. The airflow is calculated with the board in the upright position. |

1. This information is based on 8.0W (typical) measured power dissipation of the UltraSPARC IIe 500MHz CPU.
2. The maximum ambient temperature is 55°C. (Ambient temperature is the temperature of air flowing to the heatsink.
3. NEBS Level 3 requirement is 55°C at 6,000 ft. elevation for up to 96 hours.

# Bibliography

## General References

It contains the latest informat ion about the entire UltraSPARC product line, including HTML, Postscript, and PDF copies of UltraSPARC processor data sheets and manuals.

The docs.sun.com[SM] Web site enables you to access Sun technical documentation online. You can browse the docs.sun.com archive or search for a specific book title or subject. The URL is http:/ /docs.sun.com

## Books and Specifications

*PCI Local Bus Specification*, Revision 2.1: PCI Special Interest Group, Portland OR, June 1995, www.pcisig.com

*PCI Hot-Plug Specification*, Revision 1.0: PCI Special Interest Group, Portland, Oregon, USA, www.pcisig.com

*CompactPCI Specification* - PICMG 2.0 R3.0, October 1, 1999, PCI Industrial Computers Manufacturers Group

*CompactPCI Specification Short Form* - PICMG 2.0 R2.1, September 2, 1997, PCI Industrial Computers Manufacturers Group

*CompactPCI Hot Swap Specification* - PICMG 2.1 R1.0, 3 August 1998, PCI Industrial Computers Manufacturers Group, Wakefield, MA

*CompactPCI Multi Computing* Specification - PICMG 2.14; This reference covers several documents presently in draft form

*CompactPCI Computer Telephony Specification* - PICMG 2.5 R1.0, April 3, 1999, PCI Industrial Computers Manufacturers Group

*CompactPCI Power Interface Specification* - PICMG 2.11 R1.0, October 1, 1999, PCI Industrial Computers Manufacturers Group

*PMC on CompactPCI Specification,* PICMG 2.3 R1.0, August 7, 1998, PCI Industrial Computers Manufacturers Group

*CompactPCI â 6U Dual System Slot Specification*, PICMG 2.7, PCI Industrial Computer Manufacturers Group

IP on *CompactPCI Specification,* PICMG 2.4 R1.0, August 7, 1998, PCI Industrial Computers Manufacturers Group

*PCI-PCI Bridge Board Connector for Single Board Computer*, PICMG 1.1 R1.0, PCI Industrial Computers Manufacturers Group

*VME64X on CompactPCI Specification* - PICMG2.2 R1.0, August 7, 1998, PCI Industrial Computers Manufacturers Group

*Keying of CompactPCI Boards and Backplanes - PICMG 2.10 Draft 0.4*, September 28, 1998, Industrial Computers Manufacturers Group

*H110 Hardware Compatibility Specification: CT Bus*, Revision 1.0, 1997, Enterprise Computer Telephony Forum

*Intelligent Platform Management Interface Specification*, v1.0, Document Revision 1.1, 26 August 1999, Intel Corporation, Hewlett Packard Company, NEC Corporation, Dell Computer Corporation

*Intelligent Chassis Management Bus Bridge Specification*, v1.0, Document Revision 1.00, 26 August 1999, Intel Corporation, Hewlett Packard Company, NEC Corporation, Dell Computer Corporation

*EuroBoard Specification*, IEC 297-3 and -4, Bureau Central de la Commission Electrotechnique Internationale, Geneva, Switzerland

*Draft Specification for 2 mm Connector Systems*, IEC-61076-4-101, International Electrotechnical Commission, American National Standards Institute, New York, NY, USA

*Draft Standard for a Common Mezzanine Card Family: CMC, P1386 Draft 2.1*, IEEE, New York, NY Oct. 1999—covers mechanical specifications for PMC cards

*PMC I/O Module Standard, VITA 36, 199X*, Draft 0.1, July 19, 1999, VITA Standards Organization, Scottsdale, AZ—gives mechanical definition and contact assignments for PIM cards

*Draft Standard Physical and Environmental Layers for PCI Mezzanine Cards PMC, P1386.1 Draft 2.1*, IEEE, New York, NY Oct. 1999—covers electrical specifications and contact assignments for PMC cards

*IEEE Standard for Mechanical Rear Plug-in Units Specifications for Microcomputers Using IEEE 1101.1 and IEEE 1101.10 Equipment Practice*, IEEE Std 1101.11-1998, IEEE, New York, NY, 1998. ISBN 0-7381-0179-6

IEEE 1101.1-1991, IEEE Standard for Mechanical Core Specifications for Microcomputers Using IEC 603-2 Connectors, IEEE, New York, NY

*IEEE Standard for Additional Mechanical Specifications for Microcomputers Using the IEEE Std 1101.1-1991 Equipment Practice*, IEEE Std. 1101.10-1996, IEEE, New York, NY, 1997. ISBN 1 55937-863-8

*IEEE Standard for Additional Mechanical Specifications for Microcomputers using IEEE 1101.1 Equipment Practice*, IEEE 1101.11, Institute of Electrical and Electronics Engineers, Inc., 445 Hoes Lane, P.O. Box 1331, Piscatoway, NJ, USA, 08855-1331

*IEEE Standard: Test Access Port and Boundary-Scan Architecture*, IEEE Std 1149.1-1990, IEEE, New York, NY, 1990.

IEEE Standard: *Telecommunications and Information Exchange Between Systems--Local and Metropolitan Area Networks--Specific Requirements--Part 3: Carrier Sense Multiple Access With Collision Detection (CSMA/CD) Access Method And Physical Layer Specifications*, IEEE Std 802.3–1998: IEEE, New York, NY, 1998. (This edition includes all contents of the 8802-3:1996 Edition, plus IEEE Std 802.3aa-1998, IEEE Std 802.3r-1996, IEEE Std 802.3u-1995, IEEE Std 802.3x&y-1997, and IEEE802.3z-1998).

*21554 PCI-to-PCI Bridge for Embedded Applications* Product Preview Datasheet, Order Number: 278089-001 Dec 1998, Intel Corporation

*21554 PCI-to-PCI Bridge for Embedded Applications Hardware Reference Manual* Order Number: 278091-001, September 1998, Intel Corporation,

*21154 PCI-to-PCI Bridge Configuration Application Note* Order Number: 278080-001, October 1998, Intel Corporation

*21554 Embedded PCI-to-PCI Bridge Hardware Implementation Application Note* Order Number: 278218-002, March 1999, Intel Corporation

Linear Technology LTC1643L/LTC1643H PCI-Bus Hot Swap Controller, September 1998

*DS80CH11 System Energy Manager Product Specification*, V2.0 15. Dallas Semiconductor

*High-Speed Microcontroller User's Guide*, Dallas Semiconductor

$I^2C$ Peripherals Data handbook IC12, Philips Semiconductors, 1997

Mindshare, Inc: Shanley, Tom, and Don Anderson. *PCI System Architecture* , 4th ed., Reading MA: Addison Wesley, 1999

Solari, Edward, and George Willse. *PCI Hardware and Software Architecture and Design*. San Diego: Annabooks, 1998.

# Sun Microsystems Publications

These books and papers are available in printed form, and some are also available through the World Wide Web. See *Preface* for information about the SME www pages.

## Solaris Operating Environment

*Solaris 8 (SPARC Platform Edition) Installation Guide, Part Number 806-0955 -10, February 2000*

*Solaris 8 Advanced Installation Guide, Part Number 806-0957-10, February 2000*

*Solaris 8 System Administration Supplement, Part Number 806-6611-xx*

*System Administration Guide, Volume 1*, Part Number 805-7228-10, February 2000 (contains chapters on device management and configuring devices—presently deals with Hot Plug considerations. (Description of Hot Swap support should appear in documentation associated with an early maintenance release of Solaris 8 Operating Environment.)

*System Administration Guide*, Volume 2, Part Number 805-7228-10, February 2000 (of general interest)

*System Administration Guide, Volume 3*, Part Number 805-7228-10, February 2000 (of general interest)

*man pages section 1M: System Administration Commands*, Part No: 806-0625-10 February 2000 (covers cfgadm command; see above discussion on Hot Swap support in the reference to the *System Administration Guide, Volume 1*.)

*OpenBoot 3.x Command Reference Manual*, Part Number 806-1377-10, February 2000, Revision 01

*Writing FCode 3.x Programs*, Part Number 806-1379-10 February 2000, Revision A

*OpenBoot 3.x Quick Reference*, Part Number 806-2908-10, February 2000, Revision A

*Solaris Naming Administration Guide, Part Number 806-1387-10, February 2000*

*Solaris Naming Setup and Configuration Guide, Part Number 806-1386-10, February 2000*

*Writing Device Drivers, Part Number 805-737810, February 2000 (includes information about the Device Tree)*

## Alternate Pathing

These documents can be read for an approximation of the installation procedure that will apply to the CP2000 HA platform.

*Sun Enterprise Server Alternate Pathing 2.3 User Guide,* Part No. 806-1933-10 February 2000, Revision A

*Sun Enterprise Server Alternate Pathing 2.3 Installation Guide and Release Notes, February 2000*

## SunVTS

*SunVTS 4.0 User s Guide,* Part Number 806-2057-10 Revision A, February 2000

SunVTS 4.0 Test Reference Manual, Part Number 806-2058-10, Revision A, February 2000

## Processors and Integrated Circuits

*Advanced PCI Bridge User s Manual*, Part Number 805-1251-01, November 1997

## CP2000 Family System Documents

- *CP2000 Family Theory of Operation* (P / N 960-1164-xx)
- *Netra CP2060 CompactPCI Read Me First* (P / N 816-0852-xx)
- *Netra CP2080 CompactPCI Read Me First* (P / N 816-0853-xx)
- *XCP2060-TRN I/O Transition Card Manual for Netra CP2060/CP2080 CompactPCI Boards* (P / N 806-6203-xx)
- *Memory Module Installation/Removal Guide for CP2000 Family CompactPCI Boards* (P / N 816-0854-xx)

# Glossary

## Introduction

The terminology used in this glossary is not necessarily in agreement with standard Sun terminology but it is in agreement with PICMG and Telco industry terminology.

## Glossary Listing

**Alternate Pathing** Alternate Pathing (AP) is a software-driven facility that employs both redundant hardware and redundant software driver paths between a server and a disk subsystem or a network. If one path fails, AP can ensure that the disk subsystem or network is still available through the alternate path. For example, the alternate path can be a second port on an interface board, or an entirely separate interface board. See also Dynamic Reconfiguration.

**Advanced System Monitoring** Advanced System Monitoring (ASM) is the provision of hardware status information to the user or application program to allow an orderly shut down to be made before a hardware failure causes any damage.

**Availability** The ratio of the total time that a functional unit is capable of being used to the total time that the unit is required for use.

**BMC** Baseboard Management Controller; the BMC is used to manage chassis environmental, configuration and service functions and receive event data from other parts of the system. It can receive data through sensor interfaces, and interprets these data by using the Sensor Data Repository (SDR) to which it provides an interface. The BMC maintains and provides an interface to the

System Event Log (SEL). The BMC allows Both the SDR and the SEL to be accessed from the system or from the Intelligent Platform Management Bus (IPMB). A typical function of the BMC is to measure processor temperature, power supply values, and cooling fan status. It can take some autonomous actions to preserve system integrity. For example, it might switch on a fan at a particular temperature threshold. An application interface may be provided to enable custom user-management applications to be built.
The BMC describes an abstract function, or role. It carries no definition of how it might be implemented.

**checkpoint** (1) A point at which information about the status of a job and the system can be recorded so that the job can later be restarted from that point. (2) A sequence of instructions in a computer program for recording the status of execution for restarting. v. to checkpoint; n. checkpointing

**CompactPCI** An adaptation of the PCI bus architecture defined in the Peripheral Component Interconnect (PCI) Specification 2.1 (or later) to an electrically-compatible robust industrial form. This form specifies an Eurocard-style circuit board assembly that uses “hard metric” connectors to connect it to the enclosure backplane. CompactPCI is an open specification supported by the PCI Industrial Computers Manufacturers’ Group (PICMG).

**CompactPCI Bridge** The PCI bridge between the System Host processor and the CompactPCI bus. The CompactPCI bridge must reside in the system slot to provide CompactPCI clocking and arbitration that are only available from that slot. CompactPCI Bridges must be controllable by the System Management Controller to turn off clocks and arbitration.

**Device Reconfiguration** A process that is used in the CP2000 system to configure (add) or deconfigure (remove) Device Tree allocations and load or unload software driver modules while the system is running. It is analogous to *Dynamic Reconfiguration* that is used on some Sun high-end server systems with the important differences: it is not used to reconfigure memory or CPU resources and it can be used automatically in the Full Hot-Swap and HA Hot-Swap cases when the Hot-Swap framework software is prompted by the System Management Controller. CP2000 HA device reconfiguration can also be invoked manually from a console.

**Domain** That part of a computer network in which the data processing resources are under common control. See *PCI Domain*.

**Dynamic Reconfiguration** Dynamic Reconfiguration (DR) is a software package that enables the administrator to (1) view a system configuration; (2) suspend or restart operations involving a port, storage device, or board; and (3) reconfigure the system (detach or attach hot-swappable devices such as disk drives or interface boards) without the need to power down the system. When DR is used with Alternate Pathing or Solstice DiskSuite software (and redundant hardware), the server can continue to communicate with disk drives and networks

without interruption while a service provider replaces an existing device or installs a new device. DR supports replacement of a CPU/Memory, provided the memory on the board is not interleaved with memory on other boards in the system. Note that DR is used with Sun high-end server systems. See *Device Reconfiguration* for the analogous process that is applied to CP2000-based systems.

**Dropin** A Dropin is a code or data module which can be called by the OBP during system startup. It is placed in unused memory space between OBP and POST. User-created dropins are usually used to initialize custom user hardware. They do not require that the user possesses OBP source code; only the binary OBP image need be licensed. Dropins are used to add firmware drivers for user hardware

**Failover** The process of transfer of function from a failed component subsystem to an alternate one while preserving the operational state of the overall system. The functions transferred may include those of control and management.

**Firmware** An ordered set of instructions and data that is stored in a way that is functionally independent of main storage, for example, microprograms stored in a read-only memory (ROM). The term *firmware* describes microcode in ROM. At the time they are coded, microinstructions are software. When they are put into ROM they become part of the hardware (microcode) or a combination of hardware and software (microprograms). Usually, microcode is permanent and cannot be modified by the user but there are exceptions.

**FRU** Field Replaceable Unit: a part or subsystem that may be replaced in the field or at a customer-site. Parts that are not FRUs are only factory replaceable.

**GMII; Gigabit Media-independent Interface** The GMII is an Ethernet network specification that defines a standard 1000-megabit interface between the MAC layer and either of the physical layers: 1000BASE-X (fiber-channel family) or 1000BASE-T (UTP). It accommodates these physical layers without having to modify the upper layers (that is, the protocol stacks) for the particular transmission medium. It is defined in IEEE Std. 802.3z–1998, which is included in IEEE Std. 802.3–1998.

**Handover** Synonymous with *switchover*; The process of transfer of function from a component subsystem to an alternate one while preserving the operational state of the overall system. The functions transferred may include those of control and management. Handover occurs when there is no failure in the system to prompt the transfer (compare *failover).*

**Heartbeat** A repetitive signal passed from one system to another to communicate the state of integrity or “health” of the sending system

**High availability** High Availability describes the property of a system associated with a high in-service to out-of service time ratio. This property may be engineered by reconfiguring the system “on the fly” to isolate failed elements so they can be replaced without affecting the operational condition.

**Host Computer** Host Computer (Context dependent): (1) A computer that usually performs network control functions and provides end-users with services such as computation and database access.
(2) The primary or controlling computer in a multi-computer installation.

**Hot Plug** A slot must be powered down and isolated from the bus before an Adapter Card can be inserted. The Hot Plug specification requires that board power be controlled and that means be provided to set or maintain the board in a quiescent state prior to its insertion or removal.

The method of putting the board in a quiescent state or of controlling power application to it is not defined in Hot plug but is left to the system manufacturer. The Hot Plug Interface is defined by the PCI SIG—see the Hot-plug specification in the Bibliography.

**Hot Swap** The capability or property of a system element to be removed or replaced while the system hardware is nominally operating under power. This capability is usually invoked after a failure and is implemented by a sequence that steers the functions of the element to other parts of the system.

Hot Swap, as defined by PICMG, can be classified as Basic, Full, or HA.

Basic Hot Swap requires manual software sequencing to bring a card out of commission.

Full Hot Swap uses hardware enumeration signals to indicate board status. Software automatically decommissions the card.

HA Hot Swap provides for a fully automated decision tree and use of software and a System Management Controller / Hot-Swap Controller to decommission or commission a card.

**Hot-Swap Controller** The controller that takes care of the low-level sequencing associated with Hot Swap.

**ICMB** Inter-chassis Management Bus: an IPMI / $I^2C$ bus (analogous to the IPMB) used to accomplish chassis-to-chassis management.

**IHB** Inter-Host Bus; an IPMI / $I^2C$ bus (is the IPMB) used for direct communications between controllers on host boards.

**IO** Input / output; applies to system peripheral signals

**IPMB** Intelligent platform management bus; a bus that carries serial communication signals that comply with the IPMI; it is used to communicate between CompactPCI circuit boards in a chassis.

**IPMI** Intelligent Platform Management Interface: IPMI is a protocol interface with a protocol stack that includes link, transport and session layers to provide reliability. It resides on an $I^2C$ physical layer.

**$I^2C$** Inter-Integrated Circuit Bus: a serial bus developed by Philips for inter-package communications and typically used by them in TV sets. In Sun CompactPCI systems, it is used to link card elements in a system for management communications.

**$I_20$** Intelligent IO; a messaging protocol associated with the 21554 PCI to PCI bridge.

**KCS interface** Keyboard Control Style interface; This interface is defined in the IPMI Specification (See Bibliography). It is one of the BMC to System Management Software (SMS) interfaces.

**LVD SCSI** A version of the SCSI bus that uses LVTTL (3.3 V) differential logic technology; this bus is currently specified with an 80 MHz maximum transfer rate and a maximum cable length of 18 ft.

**MAC Address** Medium Access Control address; synonymous with Ethernet address. The MAC address is a 48-bit address used to direct data-link layer transactions.

**MII Media-independent Interface:** The MII is a specification that defines a standard interface between the MAC layer and any of the three physical layers: 100BASE-TX, 100BASE-T4 or 100BASE-FX). It can support both 10 Mbps and 100 Mbps data rates. Since the electrical signals are clearly defined, the MII may be implemented internally or externally in a network device.

**Nexus** (from *nectere,* to bind: a connection between individuals of a group); In this context, a Nexus driver supports a bridging connection for communication between devices on separate buses. These devices may be arranged in a hierarchal tree configuration with a number of bridges. In this case a Nexus driver is associated with each bridge to handle communications with adjacent levels in the hierarchy.

**Nines** Used as a measure of system availability; three nines > 99.9% availability, four nines > 99.99%, five nines > 99.999%; six nines > 99.9999%; ...

**NMI** Non-maskable interrupt

**Node** An addressable point on a network. Each node in a Sun network has a different name. A node can connect a computing system, a terminal, or various other peripheral devices to the network.

**NTB** Non-transparent Bridge (21554)

**OBP** Open-boot prompt; open-boot program; open-boot PROM (context dependent). The open boot PROM, or system PROM, contains code to run POST and a suite of user-accessible subsystem hardware tests. It has a Forth interpreter for custom user routines. Under a normal boot sequence, it provides a path to a system boot device which is accessed after POST completes. “Open Firmware” is controlled by IEEE Standard 1275.

**PCI Domain** The functional entity that includes a host—usually with a host PCI bridge—and the peripherals that it controls. The domain does not necessarily uniquely include the PCI bus because this bus can be shared by multiple domains. For example, a second domain can comprise a second host/bridge element that controls a different set of peripherals on a shared bus. Separation and management of the domains is implemented by a controlling system mechanism that guarantees their mutual protection.

**Peripheral Host** see *Satellite Host*

**PMC** PCI Mezzanine Card. A PMC card fits into a special PCI-bus connector designed to attach compact peripherals. These peripherals may decode a variety of IO functions from this bus.

**POST** Power-on Self Test: a suite of tests run out of system firmware before any other code is loaded. The purpose of such testing is to check the integrity of the hardware before loading a software system.

**PICMG** PCI Industrial Computers Manufacturers' Group

**RARP** Reverse Address Resolution Protocol; The protocol broadcasts a MAC (ethernet) address and receives an IP address in response from a RARP server.

**RAS** Reliability, Availability and Serviceability; the general concepts associated with high in-service time systems and their simplicity of maintenance.

**Reliability** The ability of a functional unit to perform a required function under stated conditions for a stated period of time.

**Remote Management** The action of managing a system or group of systems from a physically distant location. Remote management of Sun systems may be performed using the *Sun Management Center* application.

**Satellite Board** see *Satellite Host*

**Satellite Host** Synonymous with *Peripheral Host* and *Satellite Board*; A Satellite Host performs independent tasks in response to commands from the system host. The Satellite Host has no accessible PCI space and is limited to controlling its own on-board IO.

**Segment** The extent to which a backplane and cards combination can be extended by accounting for signal loading. In CompactPCI, a segment spans a maximum of eight card slots, beyond which some bridge elements (system bridge) are needed to provide expansion into another segmental.

**SEL** System Event Log: the database of measured values and events that is created by the BMC based upon its sensor monitoring. This database resides in the host and is accessible by high-level applications.

**Serviceability** The capability of performing effective problem determination, diagnosis, and repair on a data-processing system.

**SDR** Sensor Data Repository: the database that the BMC uses to determine what sensors, FRU devices, and management controllers are in the system. This database contains an account of sensor locations, properties, and associations.

**Shelf** A single physical computing system composed of one or more CompactPCI bus segments. Electrical limitations of the bridging interfaces may require that the segments be in close proximity.

**Switchover** see *Handover*

**System Board Computer** The System Board Computer is the processor board that connects to a backplane system slot.

**System Host** A system host accepts interrupts and owns peripherals. It executes user applications and decides the distribution of tasks within a system. In Hot-Swap systems the system host acts as a traffic router and functions to activate and deactivate peripheral cards (plug-in boards). It is not a CompactPCI requirement that the system host reside in a system slot although this is normally the case. If it resides in a peripheral slot that slot must be wired to receive peripheral interrupts from the backplane.

**System Management Bus** A serial bus that carries data and control signal between System Management Controllers on peripheral boards and devices; Communications on this bus use the IPMI protocol over an $I_2C$ hardware layer.

**System Management Controller** There is a System Management Controller (SMC) on each card in the enclosure. One of these cards either assumes control by command or takes control after negotiation with the other System Management Controllers. The System Management Controller manages peripherals to improve the availability of the system. Through the IPMB, this entity receives information on IDs of, or problems with, cards in the system and can communicate that information with other cards or with a system host via another bus. The SMC can switch the PCI bridge, PCI arbitration, and PCI clocking on or off.

**System Slot** The card location in an enclosure that provides for CompactPCI clocking and arbitration. It follows that the CompactPCI bridge, which supplies these functions, must be in the system slot.

**System-slot Bridge** provides clocks and arbitration. This device must be controllable from somewhere. It can be controlled from a controller. The system host need not reside on the same card but the card that performs the function of system host must be able to talk to the slot containing the system-slot bridge.

**Takeover** see *failover*

**TFTP** Trivial File Transfer Protocol; a simplified form of File Transfer Protocol (FTP). It contains no password checking and is typically used to boot diskless workstations and other boot clients.

# Index

## O



## P



## R



## S



## T



## U



## W



## X