UP2000
User Manual

Part Number: 51-0031-1A

Part #: **51-0031-1A**

Date: **4/7/00**

Alpha Processor, Inc.
A Samsung Company
130C Baker Avenue Extension
Concord, MA 01742
Tel: 978.318.1100
Fax: 978.371.3177

Alpha Processor, Inc. reserves the right to make changes in its products without notice in order to improve design or performance characteristics.

This document is considered to be preliminary with further revisions and corrections in process. Alpha Processor, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication or the information contained herein, and reserves the right to make changes at any time, without notice. Alpha Processor, Inc. disclaims responsibility for any consequences resulting from the use of the information included in this publication.

This publication neither states nor implies any representations or warranties of any kind, including but not limited to, any implied warranty of merchantability or fitness for a particular purpose. Alpha Processor, Inc.'s products are not authorized for use as critical components in life support devices or systems without Alpha Processor, Inc.'s written approval. Alpha Processor, Inc. assumes no liability whatsoever for claims associated with the sale or use (including the use of engineering samples) of Alpha Processor, Inc. products except as provided in Alpha Processor, Inc.'s Terms and Conditions of Sale for such products.

Trademarks

Product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

Preliminary Product Information

The information in this publication is preliminary, and is believed to be accurate at the time of publication.. API makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication or the information contained herein, and reserves the right to make changes at any time, without notice.

Revision History

Date	Rev	Description
4/07/00	51-0031-1A	**UP2000 User Manual initial review. This manual describes the UP2000-A3 product, and was modified to include the following information:** • **Firmware platform moved to Reset PAL code, Alpha SRM Console and Alpha Diagnostics** • **COM1 port reserved for serial console devices** • **Add remote reset capabilities** • **Add thermal sensitivity information**

Table of Contents

List of Figures

List of Tables

Preface

Overview

This manual describes the Alpha Processor, Inc. UP2000 product, including the UP2000 Motherboard and the Alpha Slot B Module.

The document emphasizes the Alpha System Reference Manual (SRM) Console firmware user interface.

Topics include a description of how to:

- Install an operating system
- Check or change system configurations
- Troubleshoot basic system problems

Hardware-oriented topics include how to:

- Set system speed and L2 cache size
- Configure the memory subsystem
- Cable the I/O connections
- Cable the diskette, IDE disk, and SCSI I/O ports

Audience

This manual is intended for technicians and engineers who support resellers, dealers, system integrators and OEM vendors who supply UP2000-based systems.

Scope

This manual describes the features, configuration options, functional operation, troubleshooting analysis and user interface of the system and its Alpha System Reference Manual (SRM) Console firmware. It is a part of the Alpha Processor, Inc.'s UP2000 document set that includes the *UP2000 Quick Start Installation Guide* (51-0030-1A).

Manual Organization

The *UP2000 User Manual* is organized as follows:

- Chapter 1, “Introduction,” presents the product features and includes a functional block diagram of the system.
- Chapter 2, “System Configuration,” provides a pictorial layout of the UP2000 Motherboard with its key components. Configuration elements include jumper settings for system speed and cache memory size. Main memory guidelines, I/O disk port cabling, and non-keyed I/O connections are included.
- Chapter 3, “Electrical, Environmental and Physical Data,” furnishes the electrical and environmental requirements, and physical board dimensions.
- Chapter 4, “Software Support,” describes the firmware that forms the UP2000 user interface. Topics include installing and upgrading an operating system, initializing and upgrading the firmware, as well as displaying and configuring system parameters.
- Chapter 5, “Troubleshooting,” discusses solutions for hardware and software problems encountered during system startup.
- Appendix A, “Support, Products and Documentation,” provides directions for obtaining additional product information and technical support.

Conventions and Definitions

This section defines product-specific terminology, abbreviations, and other conventions used throughout this manual.

Acronyms

The following is a list of the acronyms used in this document and their definitions.

Abbreviation	Meaning
AGP	**Accelerated Graphics Post**
API	Application Program Interface
BIOS	**Basic Input/Output System**
CE	European Conforming
CPU	**Central Processing Unit**

Abbreviation	Meaning
cUL	Canadian Underwriters Laboratory
DIMM	**Dual Inline Memory Module**
DMA	Direct Memory Access
DRAM	**Dynamic Random Access Memory**
ECC	Error Correcting Code
EIDE	**Enhanced Integrated Device Electronics**
EMI	Electromagnetic Interference
FDD	**Floppy Disk Drive**
FIFO	First In, First Out
FPGA	**Field Programmable Gate Array**
HDD	Hard Disk Drive
IDE	**Integrated Device Electronics**
ISA	Industry Standard Architecture (re: I/O Bus)
LED	**Light Emitting Diode**
LVD	Low Voltage Differential
LW	**Late Write**
OEM	Original Equipment Manufacturer
PAL	**Privileged Architecture Library**
PCI	Peripheral Component Interconnect
PIO	**Programmed Input/Output**
PLL	Phase Locked Loop
PWB	**Printed Wiring Board**
RAM	Random Access Memory
ROM	**Read Only Memory**
SCSI	Small Computer System Interface
SDRAM	**Synchronous Dynamic Random Access Memory**
SRAM	Static Random Access Memory
SRM	**System Reference Manual**
SROM	Serial Read-only Memory
SSRAM	**Synchronous SRAM**
TIG	TTL Integrated Glue Logic
UART	**Universal Asynchronous Receiver Transmitter**
UL	Underwriters Laboratory
USB	**Universal Serial Bus**

Chapter 1 Introduction

This chapter provides an overview of the UP2000 product consisting of a UP2000 Motherboard and one or two Alpha Slot B Modules.

1.1 Features

Table 1-1 provides a summary of the UP2000 product features.

Table 1-1 UP2000 Product Features

Feature	Description	Manufacturer
Physical Form Factor:	**ATX Extended (12 inch X 13 inch)**	
Daughter Card Interface:	Supports one or two Alpha Slot B Modules using the Alpha 21264 microprocessor, at speeds of 667 or 750 MHz	Alpha Processor, Inc.
Chipset:	**21272 (Tsunami)—One Cchip, four Dchips, and two Pchips provide the following:**	**Compaq**
	• Maximum 83 MHz system bus with Double Data Rate (DDR) transfers, maximum bandwidth of 2.67 GBytes / second	
	• **One 256-bit memory bus**	
	• One 64-bit, 33 MHz PCI bus with two 64-bit slots and one 32-bit slot, and one 32-bit, 33 MHz PCI bus with three 32-bit slots	
Cache:	**External Bcache with 128-bit data path for an Alpha Slot B Module supports:**	
	• 2MB or 4 MB cache per 667 MHz processor	
	• **4MB or 8 MB cache per 750 MHz processor**	
	• Late Write (LW) Synchronous SRAMS (SSRAMs)	
Main Memory:	• **Eight 168-pin Dual Inline Memory Module (DIMM) sockets, up to 2 GB (256 MB per DIMM)**	
	• Supports Phase Locked Loop (PLL) or Register-based Synchronous Direct Random Access Memory (SDRAM) Serial Presence Detect (SPD) modules of 64 MB, 128 MB, and 256 MB	
	• **Low Voltage Transistor-Transistor Logic (LVTTL) compatible memory I/O**	
Power:	ATX power connector, supplying +3.3 Vdc, ± 5 Vdc, and ± 12 Vdc	

Table 1-1 UP2000 Product Features (Continued)

Feature	Description	Manufacturer
On-board I/O:	• **CY82C693UB Peripheral Component Interconnect (PCI)/Industry Standard Architecture (ISA) Bridge (PCI Local Bus Specification Revision 2.1 compliant), Enhanced Integrated Device Electronics (EIDE) controller, and two Universal Serial Bus (USB) ports**	**Cypress**
	• AIC-7891 Small Computer System Interface (SCSI) Controller (Ultra2 SCSI), connections for up to 15 SCSI devices	Adaptec
	• **FDC37C669 Super Input/Output (I/O) Controller—2.88 MB Floppy Disk Controller (FDC), Parallel port, two NS16C550-compliant Serial ports, Real-time Clock (RTC) port, Keyboard, and Mouse**	**SMC**
I/O Slots:	• Four 32-bit, 33 MHz PCI Slots, PCI Local Bus Specification Revision 2.1 compliant	
	• **Two 64-bit, 33 MHz PCI Slots, PCI Local Bus Specification Revision 2.1 compliant**	
	• One ISA Slot (Shared)	
Firmware:	**Embedded Alpha System Reference Manual (SRM) Console**	
System Management (via PCI-ISA Bridge I^2C Controller PCF8584):	• Monitoring of processor thermal state	**Philips**
	• **Detection of processor asset record**	

1.2 System Components

The functional components of the UP2000 are shown in Figure 1-1 in block diagram form.

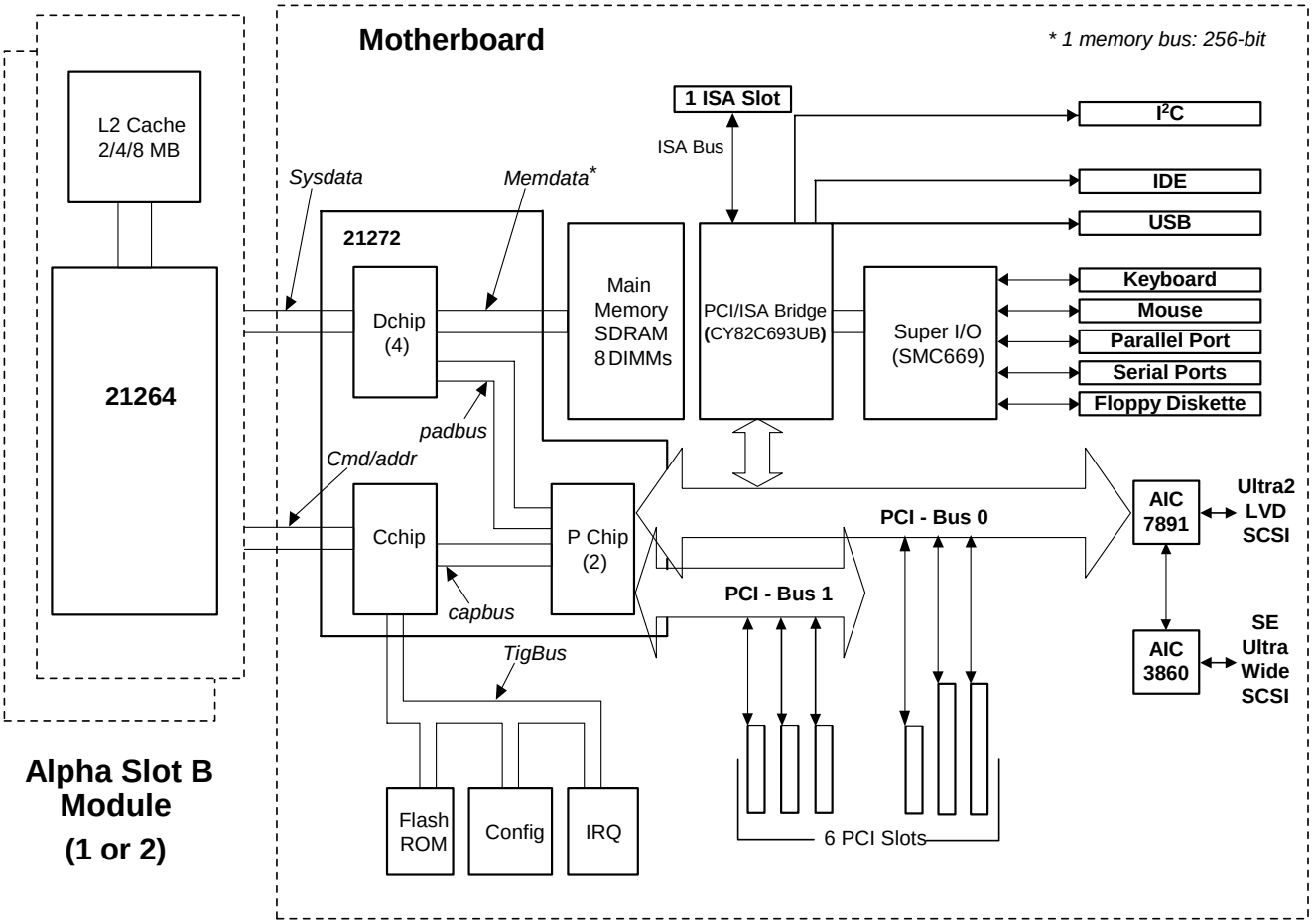

Figure 1-1 UP2000 Functional Block Diagram

Chapter 2 System Configuration

This chapter describes the layout and configuration of the UP2000 components.

2.1 Board Layout and Components

The UP2000 Motherboard uses onboard connectors to provide for:

- Alpha Slot B Modules
- Memory modules
- PCI and ISA modules
- I/O peripherals (keyboard, mouse, serial, parallel, and USB)
- SCSI devices
- IDE and floppy disk devices
- System power
- I/O connections (indicators and buttons)

These connectors and the system's configuration jumper blocks are shown in the following figure and table.

Jumper Configuration

Jumper blocks with manually inserted jumpers are used to select two system parameters:

- Speed—to match CPU rating of Alpha Slot B Module
- Size—to match L2 cache memory of Alpha Slot B Module

See section 2.2 for jumper configuration details.

Memory Configuration

Memory modules employed in the system must be selected and utilized under guidelines shown in section 2.3.

Disk Port Cabling

Disk ports (including sockets for diskette, IDE disk, and SCSI devices) are shown in Figure 2-5.

Internal I/O Cabling

Internal I/O connections for system I/O functions are shown in Figure 2-6. These connectors are not keyed, but ground pin orientation is depicted.

Rear Panel I/O Cabling

Note: *COM1 is reserved as a serial console device. All other serial devices must be connected to COM2 or to expansion serial ports. The RS232 Ring Indicator signal on COM1 is connected to the Reset function on the UP2000 Motherboard.*

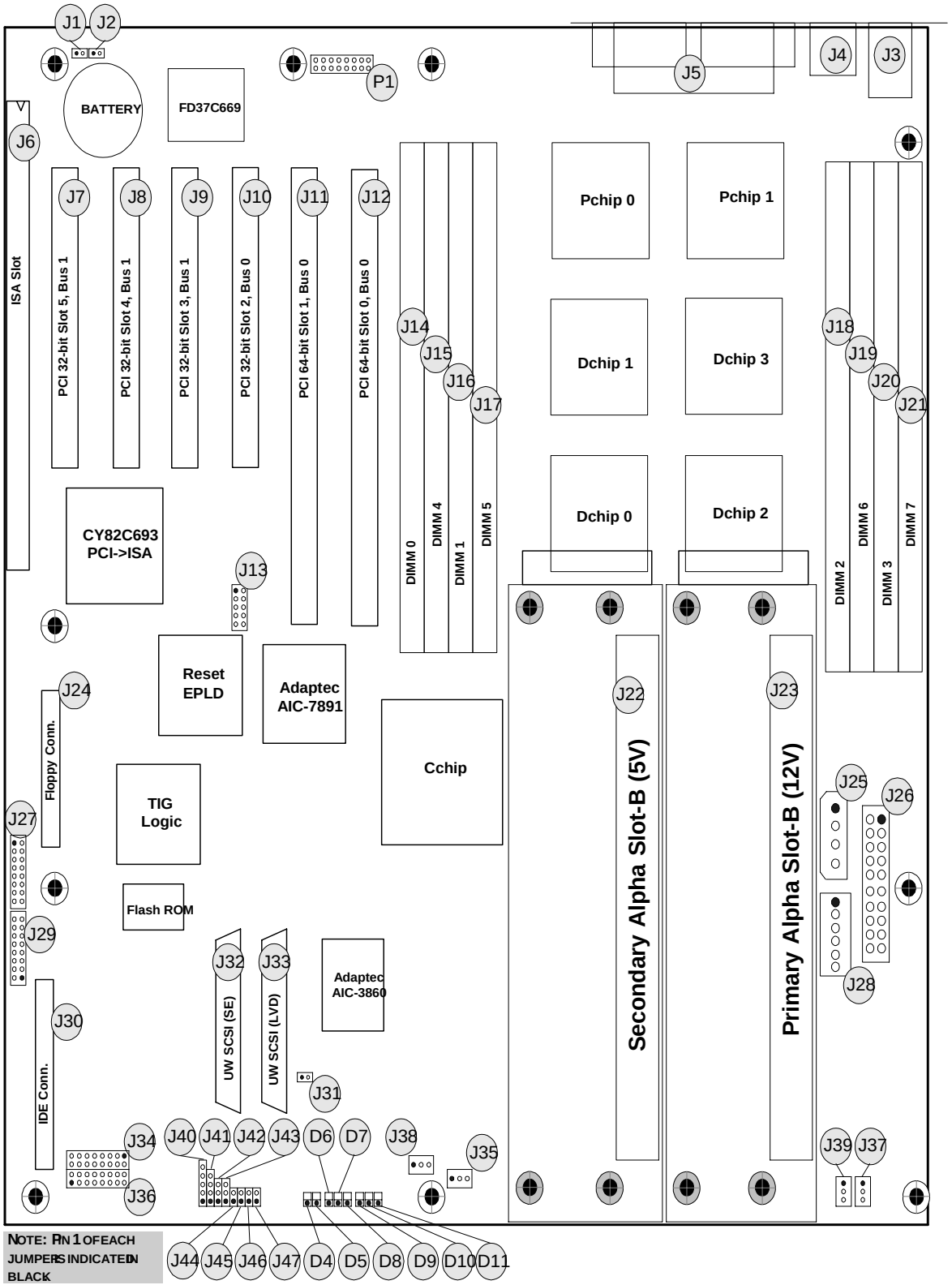

Figure 2-1 UP2000 Motherboard Layout

Table 2-1 UP2000 Motherboard Jumper and Connector Component List

Comp. No.	Specification	Comp. No.	Specification
J1	**System Chassis Fan Connector**	**J2**	**System Chassis Fan Connector**
J3	Keyboard / Mouse Connector	J4	USB Connector
J5	**D-SUB Connector (Serial IO/ Parallel IO)***	**J6**	**ISA Expansion Bus Connector**
J7	32-bit PCI Slot 5, Bus 1 Connector	J8	32-bit PCI Slot 4, Bus 1 Connector
J9	**32-bit PCI Slot 3, Bus 1 Connector**	**J10**	**32-bit PCI Slot 2, Bus 0 Connector**
J11	64-bit PCI Slot 1, Bus 0 Connector	J12	64-bit PCI Slot 0, Bus 0 Connector
J13	**Reset Electrically Programmable Logic Device (EPLD) In-system Programmability (ISP) Connector**	**J14**	**168-pin DIMM 0 Socket**
J15	168-pin DIMM 4 Socket	J16	168-pin DIMM 1 Socket
J17	**168-pin DIMM 5 Socket**	**J18**	**168-pin DIMM 2 Socket**
J19	168-pin DIMM 6 Socket	J20	168-pin DIMM 3 Socket
J21	**168-pin DIMM 7 Socket**	**J22**	**Alpha Slot B Connector for Secondary Module (5V)**
J23	Alpha Slot B Connector for Primary Module (12V)	J24	Floppy Drive Connector
J25	**Alpha Slot B Module Power Connector (4 pin)**	**J26**	**ATX Power Connector (20 pin)**
J27	System Configuration Jumper	J28	AUX ATX (Optional) Power Connector for 3.3V (6 pin)
J29	**Configuration Jumper**	**J30**	**IDE Bus Connector**
J31	Not Used	J32	Ultra-wide SCSI Single-ended (SE) Connector
J33	**Ultra-wide SCSI Low Voltage Differential (LVD) Connector**	**J34**	**CPU Speed/Flash_Sel Jumper**
J35	Alpha Slot B Module Fan Connector (Secondary)	J36	Bcache Configuration Jumper
J37	**Alpha Slot B Module Fan Connector (Primary)**	**J38**	**Alpha Slot B Module Fan Connector (Secondary)**
J39	Alpha Slot B Module Fan Connector (Primary)	J40	Power Light Emitting Diode (LED) Connector
J41	**Speaker Connector**	**J42**	**Secondary Alpha Slot B Module Debug Port Connector**

Note: ***Connector J5 includes the COM1 and COM2 ports. COM1 is reserved for a serial console device. Connect all other serial devices to COM2 or any expansion serial ports.***

Table 2-1 UP2000 Motherboard Jumper and Connector Component List (Continued)

Comp. No.	Specification	Comp. No.	Specification
J43	Primary Alpha Slot B Module Debug Port Connector	J44	Power Button Connector
J45	**Halt Button Connector**	**J46**	**SCSI LED Connector**
J47	Reset Button Connector		

Note: ***Connector J5 includes the COM1 and COM2 ports. COM1 is reserved for a serial console device. Connect all other serial devices to COM2 or any expansion serial ports.***

2.2 Configuration Jumpers

The UP2000 Motherboard has four sets of programmable jumper blocks, located at J27, J29, J34, and J36 as shown in Figure 2-2.

Alpha Slot B Module CPU speed and L2 cache size determine which configuration settings are used. Configuration setting selections are shown in Figure 2-3.

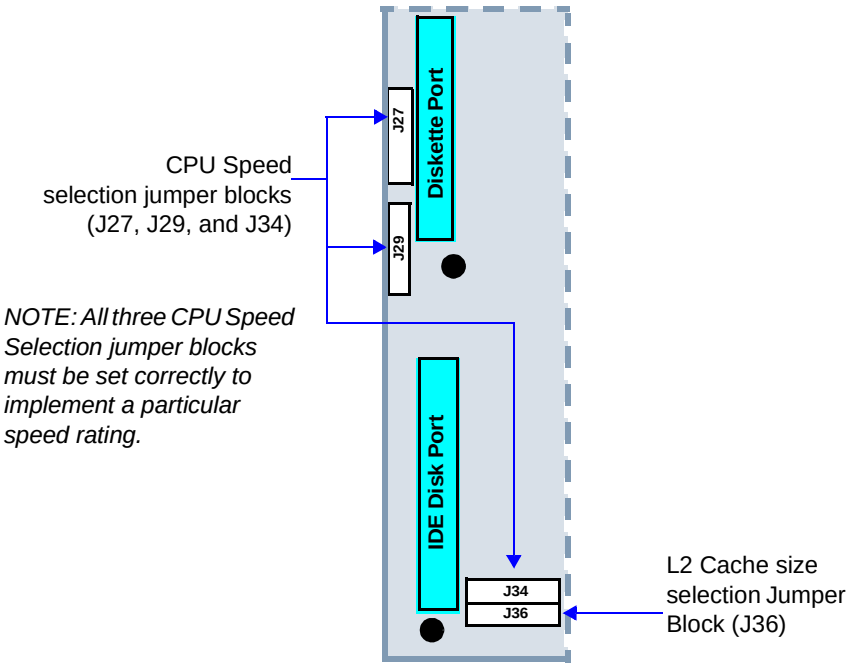

Figure 2-2 Location of Configuration Jumper Blocks

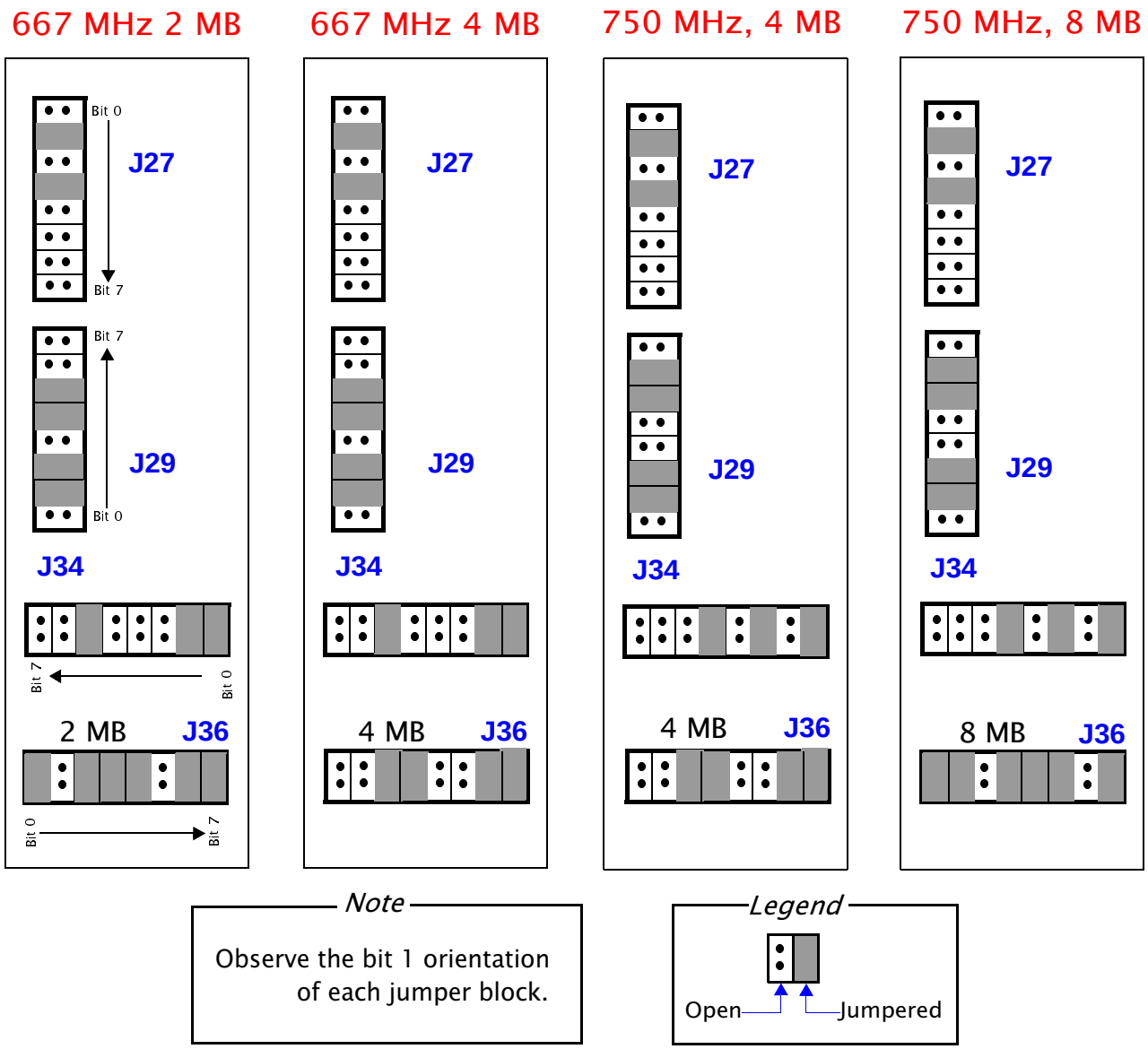

Figure 2-3 Configuration Settings

2.3 Memory Subsystem

2.3.1 Memory Configuration

The memory subsystem is comprised of two DIMM banks, designated

Bank 0 and Bank 1. Each bank has four slots (sockets) that accept 168-pin, PC100 SDRAM PLL Register based SPD DIMM modules. Slots are configured in an alternating pattern. See Figure 2-4 below.

2.3.2 Memory Guidelines

Use the following rules:

- A bank must be fully populated (all four assigned slots must be utilized).
- Populate Bank 0 first.
- A bank must utilize the same type, size, and speed DIMMs.
- Bank 0 and Bank 1 can have different type, size, and speed DIMMs.
- Memory is supported in a size range between 256 MB (minimum) to 2 GB (maximum).

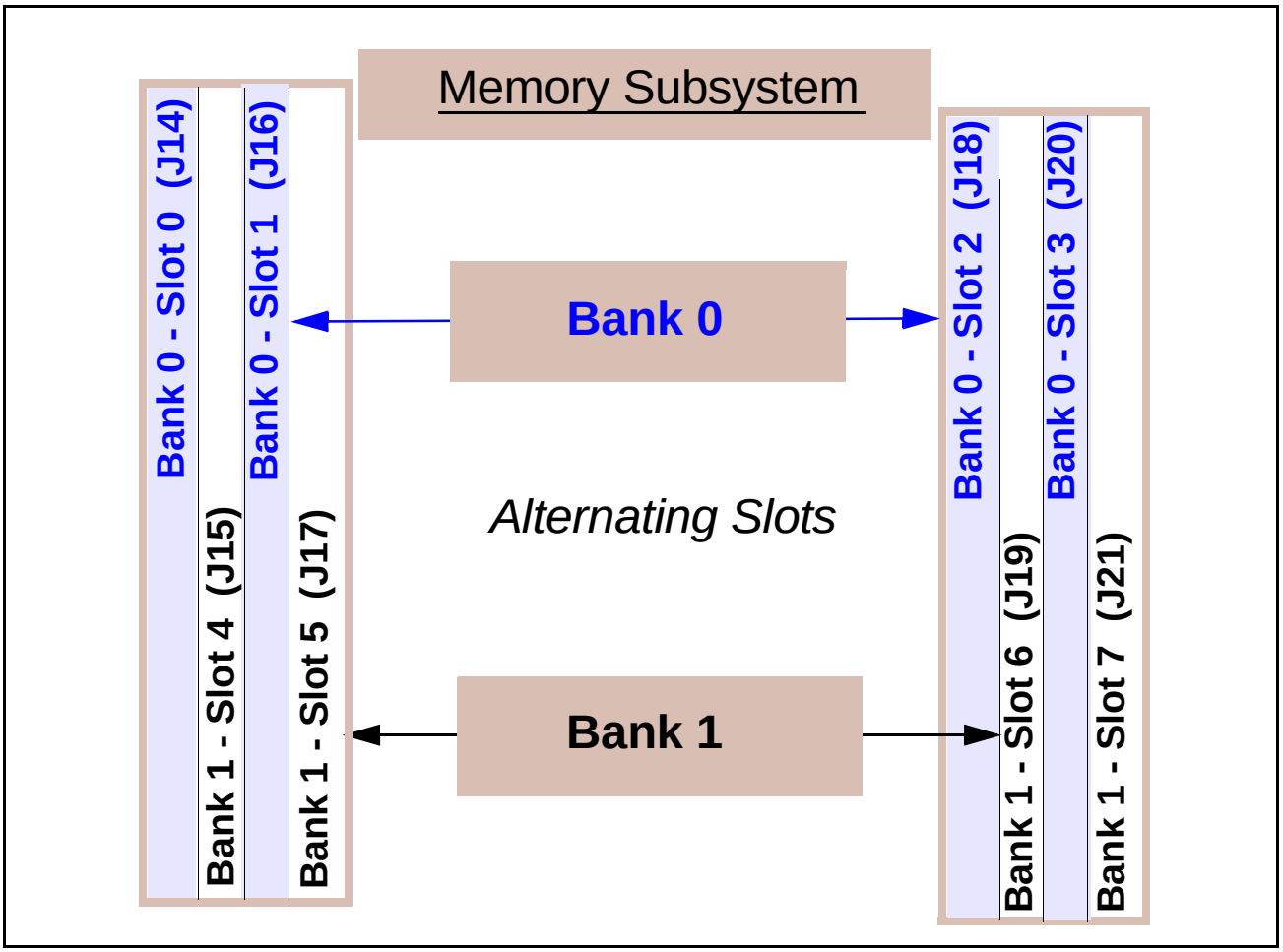

Figure 2-4 Memory Subsystem with Two Banks

2.4 Disk Ports

A pictorial view of the UP2000 disk I/O ports is shown in Figure 2-5. The IDE and diskette port sockets are keyed with one or more notches.

The SCSI connectors are high density 68-pin female sockets. For certain SCSI disk peripherals, a 50-pin to 68-pin adapter is required.

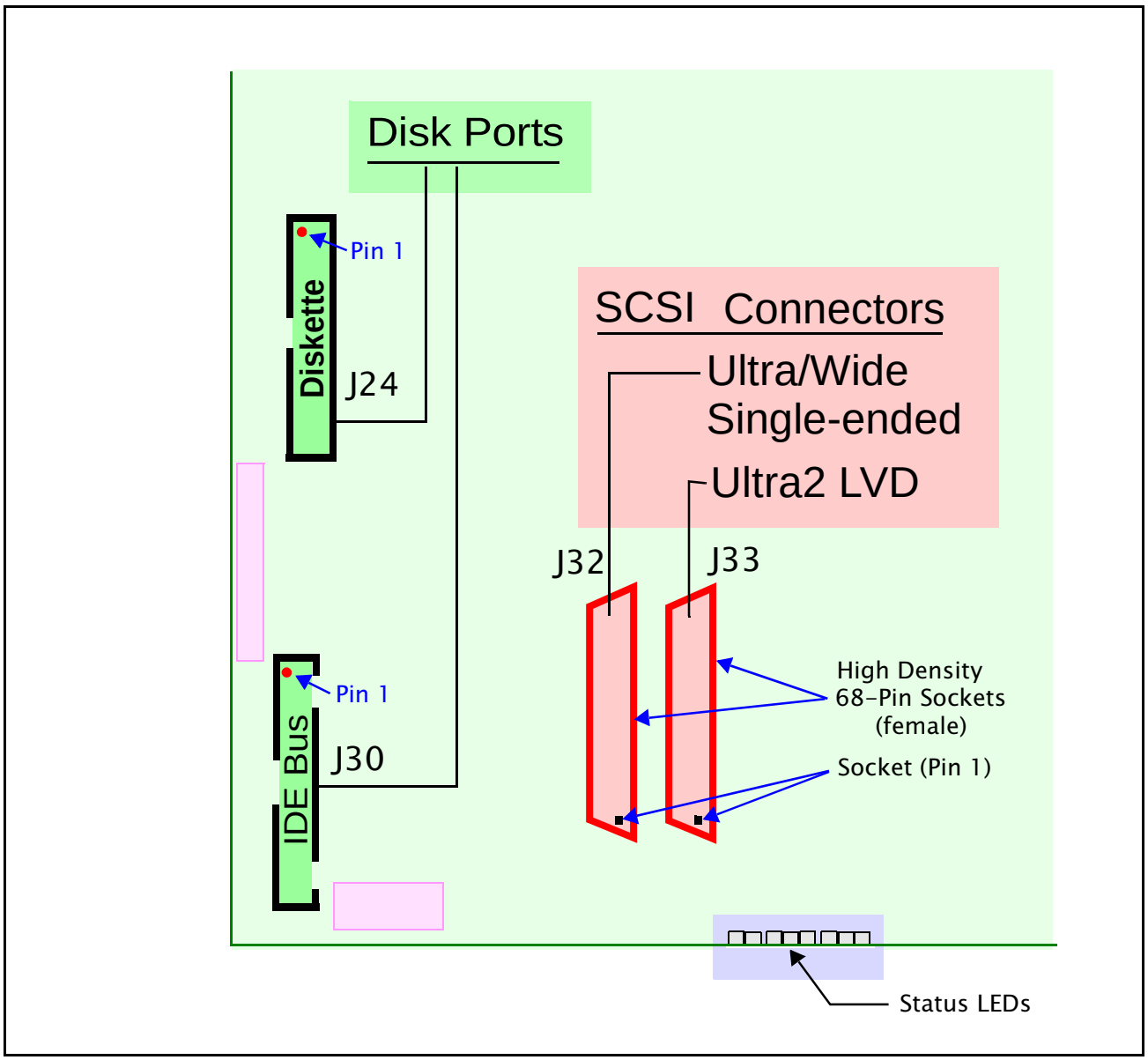

Figure 2-5 Disk Ports with Pin Orientation

2.5 Internal I/O Connections

These I/O connectors are not keyed. Figure 2-6 shows an enlargement of the connectors and the Ground position for each function.

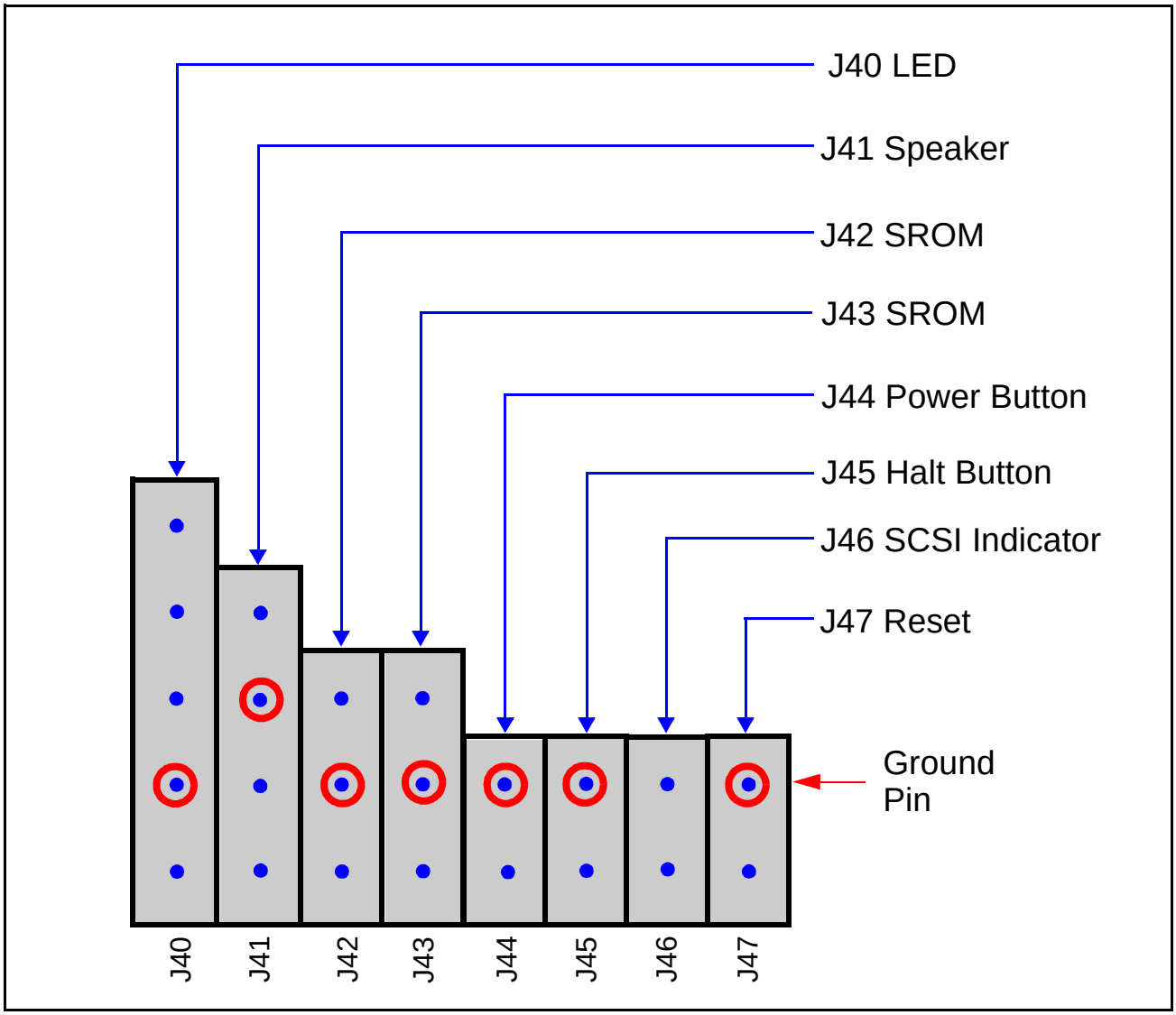

Figure 2-6 Internal I/O Connectors with Ground Orientation

Chapter 3 Electrical, Environmental and Physical Data

In this chapter, a description is provided of the UP2000 power requirements, environmental and enclosure specifications, and physical parameters.

3.1 Power Specifications

3.1.1 Power Requirements

The power connectors utilized to support the UP2000 Motherboard and up to two Alpha Slot B Modules are shown in Figure 3-1.

Note: *Connector J25 has the same form factor as power connectors used with standard disk peripherals.*

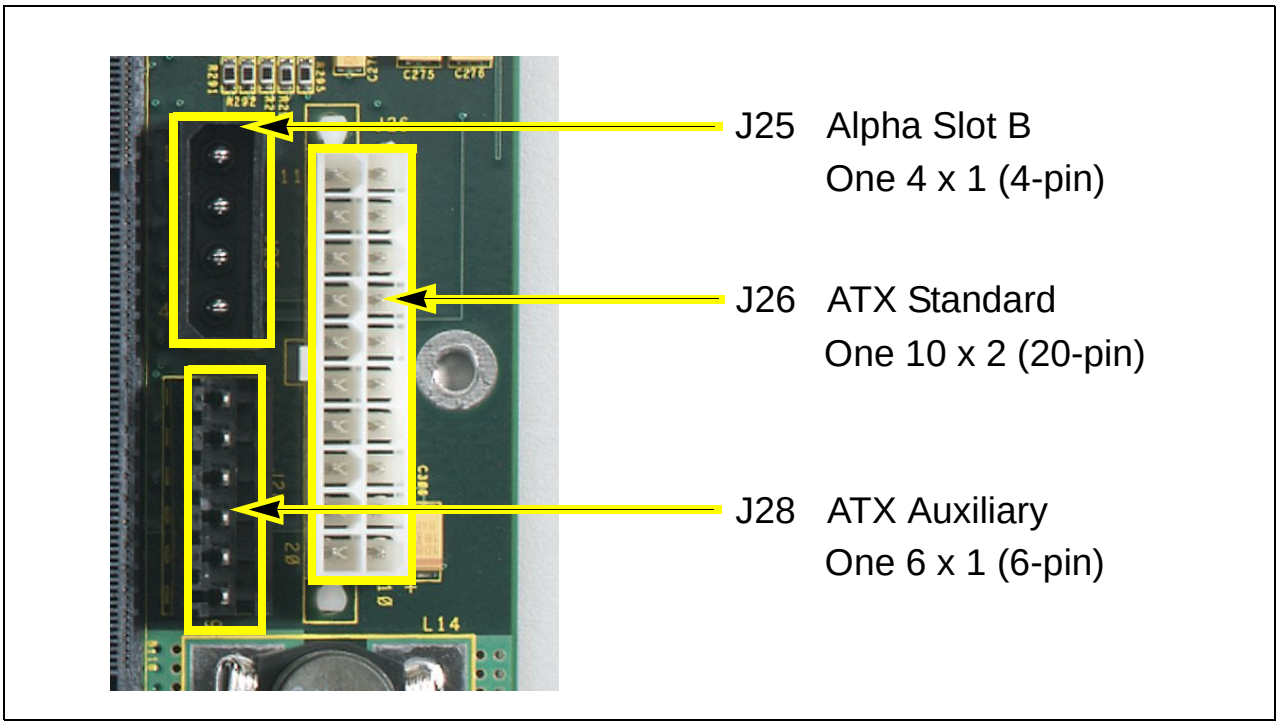

Figure 3-1 UP2000 Power Connectors

3.1.2 Estimated Power Consumption

A UP2000 system requires a 600 Watt ATX power supply. Power is distributed as follows:

- Motherboard 277W
- Alpha Slot B Module 132W (each processor)

Individual power segments are as follows:

+5V @ 70A	+12V @ 25A	+3.3V @ 50A	Total Continuous Power 500W
-5V @ 1.0A	-12V @ 1.0A	+5VSB @ 850 mA	
+ 5V and +3.3V Total Max 350W			Total Peak Power 700W

3.2 Environmental Specifications

Each Alpha Slot B Module is cooled by two small fans. Depending upon cabinetry and plug-in card requirements, additional fans for cooling the entire UP2000 system may be necessary.

The UP2000 Motherboard and Alpha Slot B Module are specified to run within the environment listed in the following table.

Table 3-1 Environmental Requirements

Parameter	Specification
Operating temperature	**5°C to 35°C (41°F to 95°F)**
Storage temperature	**-35°C to 85°C (-31°F to 185°F)**
Relative humidity	**10% to 90% with maximum wet bulb temperature 35°C (95°F) and minimum dew point 2°C (36°F)**
Rate of (dry bulb) temperature change	**11°C/hour ± 2°C/hour (20°F/hour ± 4°F/hour)**

3.2.1 Safety

The UP2000 Motherboard meets registered product-safety certification for the U.S. and Canadian Underwriters Laboratories (UL and cUL). It also meets the European Conforming (CE) standard EN 60950:1992 “Safety of Information Technology Equipment Including Electrical Business

Equipment Incorporating Amendment Nos 1, 2, 3, 4." European Norm (EN) standards which conform to the relevant directives are published in the Official Journal of the European Community.

3.2.2 EMI

The UP2000 meets electro-magnetic interference (EMI) emission certification for the following:

- Federal Communications Commission (FCC) 47 CFR Part 15 Class A (USA)
- EN 55022:1994 / A1:1995 / A2:1997 Class A ITE emissions requirements (EU)
- ICES-003 Issue 3 Class A Digital Apparatus (Canada)
- VCCI Class A ITE (Japan)
- AS / NZS 3548:1995 / CISPR 22 Class A ITE (Australia)
- SABS CISPR 22:1993 Class A ITE (South Africa)

Note: *Alpha Processor, Inc. recommends the use of high-quality, shielded cables for all I/O.*

3.3 Enclosure Requirements

Chassis or enclosures must be capable of:

- Mounting the Extended ATX form factor motherboard
- Accommodating eight goalpost assembly mounting holes

The goalpost assembly is a mechanical fixture for aligning and supporting an Alpha Slot B module. Two goalpost assemblies are shipped with the UP2000 product. Both fixtures should be installed, even if initially the system is configured for only a single processor.

3.3.1 Mounting Procedures

Chassis Mounting Holes

1. Identify the ten standard Extended ATX mounting holes.
2. Check for the eight Alpha Slot B Module goalpost mounting holes. For chassis or enclosures without this mounting hole pattern, obtain a drill template or chassis mounting tips from our website:
   http://www.alpha-processor.com/products/downloads /customer_support/UP2000.UP2000_drill_template.pdf

WARNING: ***Slot B goalposts must be securely fastened to chassis backplate.***

Refer to the* UP2000 Quick Start Installation Guide *(P/N 51-0030-1A) for complete mechanical installation details.

Motherboard & Goalposts to Chassis Assembly

Requirements for attaching motherboard and goalposts fixture to the chassis are:

- Fastening Hardware
  - Eight Alpha Slot B hex male/female spacer screws and spacer washers
  - Eight M4X6 mm long pan head Phillips screws
  - Two M3x8mm long pan head Phillips SEMS lock and flat
  - Brace for dual goalpost assembly
  - Assorted standoff, screws, and miscellaneous hardware supplied by the chassis vendor
- Tools
  - Phillips head screwdriver
  - Flat head screwdriver
  - Torque wrench
  - Nut driver (1/4")
- Proper chassis mounting holes drilled for goalpost captive nuts

Use the following mounting technique:

1. Install goalpost with fan cables labeled J37 and J39 over primary Alpha Slot B Connector. Install goalpost with fan cables labeled J35 and J38 over secondary Alpha Slot B Connector.
2. Align the goalposts fixture holes with the corresponding motherboard clearance holes.
3. Affix goalposts to motherboard from the backside using captive nuts. Torque to 8 inch/lbs.
4. Secure this assembly to the chassis with the standoffs and screws supplied by the chassis vendor.
5. Secure M4x6 mm screws through chassis to female thread of hex male/female spacer screws (eight places).
6. Attach brace to top left corners of both fan mounting plates with M3x8mm pan head Phillips SEMS lock and flat.

3.4 Physical Parameters

3.4.1 UP2000 Motherboard Parameters

The UP2000 Motherboard is a printed wiring board (PWB) with the dimensions specified in Table 3-2.

Table 3-2 UP2000 Motherboard Physical Parameters

Dimension	Value
Length	**330.2 mm (13.0 in)**
Width	**304.8 mm (12.0 in)**
Height (board only)	**1.6 mm (0.063 in)**
Total Product Height (from underside of motherboard to top of goalpost assembly)	**127.0 mm (5.00 in)**

3.4.2 Alpha Slot B Module Parameters

The Alpha Slot B Module is an assembly with the dimensions specified in Table 3-3.

Table 3-3 Alpha Slot B Module Physical Parameters

Dimension	Value
Length	**168.8 mm (6.65 in)**
Width	**47.8 mm (1.9 in)**
Height	**114.3 mm (4.5 in)**
Weight	1.0 Kg (2.2 lbs) 667 MHz
	1.2 Kg (2.7 lbs) 750 MHz

3.4.3 UP2000 Motherboard Mounting Hole Specifications

The UP2000 Motherboard mounting hole specifications are depicted in Figure 3-2. Note the goalpost assembly mounting area.

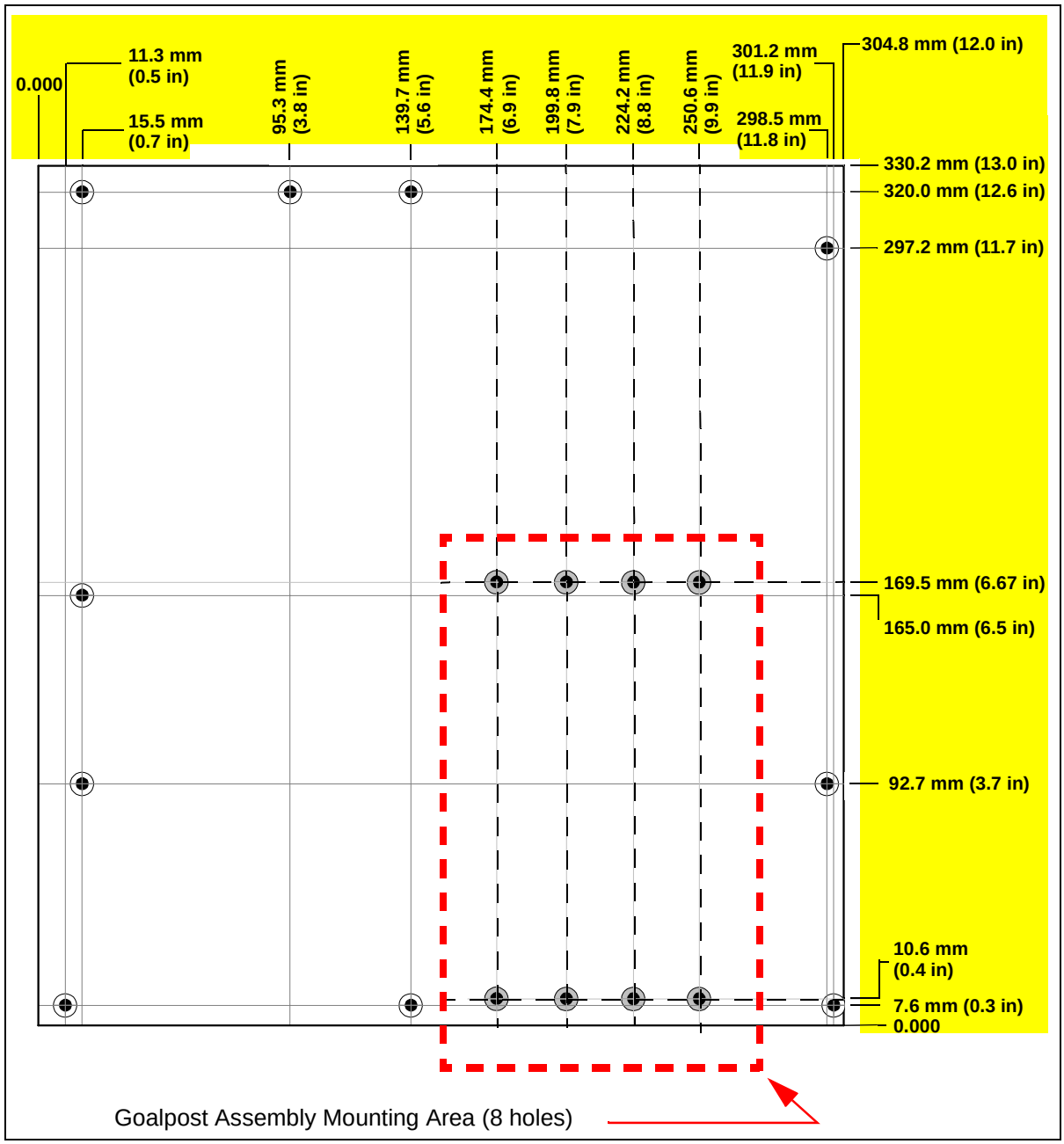

Figure 3-2 UP2000 Motherboard Mounting Hole Specification

3.5 I/O Shield information

The system support connectors must be fitted with a suitable “Venus” I/O shield. ATX-compatible motherboards have various core I/O shield designs. The UP2000 utilizes the design shown in Figure 3-3.

Individual connectors are designated with letters. Each connector type and its description are listed in Table 3-4.

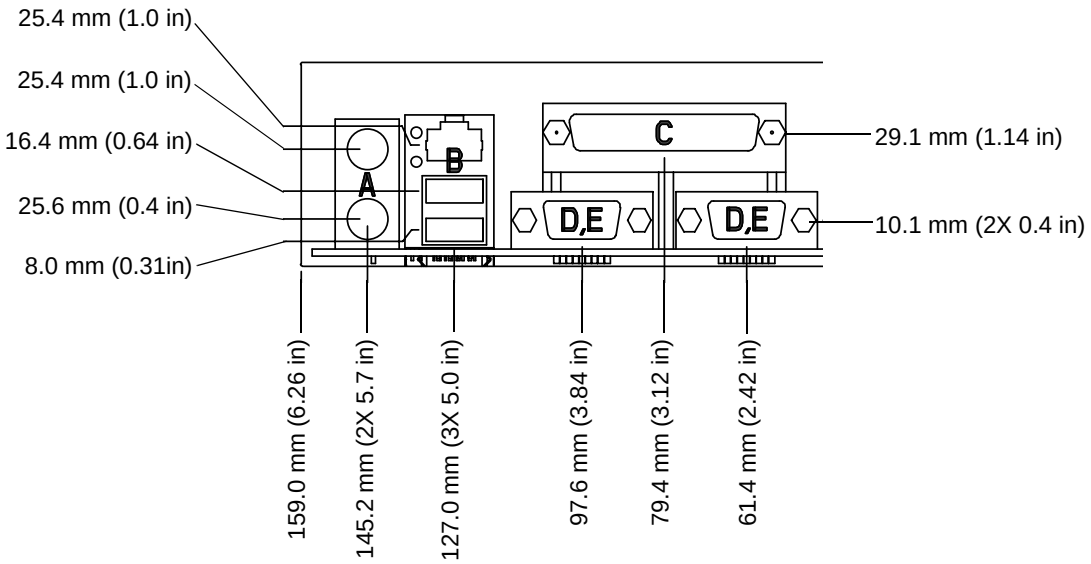

Figure 3-3 I/O Shield

Table 3-4 RearPanel I/O Connectors

Letter	Connector Description
A	PS/2 Stacked Mouse/Keyboard (DIN)
B	Dual Stack USB
C	Stacked Parallel (25 Pin D-Sub)
D,E	Serial (9 Pin D-Sub)

WARNING: ***COM1 is reserved as a serial console device. All other serial devices must be connected to COM2 or to expansion ports. The RS232 Ring Indicator signal on COM1 is connected to the Reset function on the UP2000 Motherboard.***

Chapter 4 Software Support

4.1 Software Overview

UP2000 systems support three major software components:

- Alpha System Reference Manual (SRM) Console
- Alpha Diagnostics
- Operating System

4.1.1 Alpha SRM Console

The SRM Console is firmware that initializes the UP2000 system and enables you to install and boot the operating systems. Alpha SRM Console firmware resides in the flash ROM on the UP2000 Motherboard.

For further information about the Alpha SRM Console, visit our web site at:

http://www.alpha-processor.com/support/srm.html

Navigate from the Alpha Processor, Inc. web site:

www.alpha-processor.com

and continue through the following steps:

1. Click on Support in the upper right of your browser.
2. Click on FAQs in the center left of your browser.
3. Scroll through the FAQs until you reach the one on SRM.
4. Click on the word, "here." This takes you to the SRM web pages.

Also refer to the Alpha Linux home page:

http://www.alphalinux.org

4.1.2 Alpha Diagnostics

The Alpha Diagnostics firmware is used internally by Alpha Processor, Inc. for diagnostic purposes.

4.1.3 Operating System

The UP2000 works with the Linux kernel 2.2.14 or higher in order to boot

from SRM 5.6-3 or higher.

Note: *Consult the Alpha Processor, Inc. web site for a list of OS vendors and versions currently supported.*

4.2 Alpha SRM Console

The Alpha SRM Console is the command line interface that supports the Linux operating systems. The SRM Console is used to bootstrap the operating system, configure and test the system hardware, examine system options for errors, and set or change environment variables.

This chapter describes the SRM Console commands and environment variables. Sections in this chapter are:

- Invoking the SRM Console
- Command Summary
- Displaying the System Configuration
- Booting the Operating System
- Updating Firmware
- Using Environment Variables
- Environment Variable Summary
- Finding Help

4.2.1 Invoking the SRM Console

When a system is powered up, the SRM Console runs and either remains running or passes control to an operating system. In a running system, in which control is located with the Linux operating system, shut down the operating system according to the procedure described in your operating system documentation. The SRM Console prompt, P00>>>, displays.

At the P00>>> prompt, type **boot** to return to the operating system.

4.2.2 Command Summary

The SRM Console is a command line interface, which enables you to examine and modify the system state.

Table 4-1 identifies the most commonly used SRM Console commands. Table 4-2 provides the syntax for the SRM Console commands. Table 4-3 identifies special characters you can use in SRM Console mode.

Table 4-1 Summary of SRM Console Commands

Command	Function
boot	**Loads and starts the operating system.**
clear *envar*	Resets an environment variable to its default value.
clear password	**Sets the password to zero.**
continue	Resumes program execution.
edit	**Invokes the SRM Console line editor on a RAM file or on the nvram file (power-up script).**
halt	Halts the specified processor. (Same as stop.)
help	**Displays information about the specified SRM Console command.**
initialize	Resets the system to a known state.
isacfg	**Displays or modifies parameters for ISA devices.**
lfu	Runs the Loadable Firmware Update Utility.
login	**Turns off secure mode, enabling access to all SRM Console commands during the current session.**
more	Displays a file one screen at a time.
set *envar*	**Sets or modifies the value of an environment variable.**
set password	Sets the SRM Console password for the first time or changes an existing password.
set secure	**Enables secure mode without requiring a restart of the SRM Console.**
show *envar*	Displays the state of the specified environment variable.
show config	**Displays the configuration at the last system initialization.**
show cpu	Displays the state of each processor in the system.
show device	**Displays a list of controllers and their devices in the system.**
show memory	Displays memory module information.
show pal	**Displays the version of the privileged architecture library code (PALcode).**

Table 4-1 Summary of SRM Console Commands (Continued)

Command	Function
show power	Displays information about the power supplies, system fans, CPU fans, and temperature.
show version	**Displays the version of the SRM Console program.**
stop	Halts the specified processor. (Same as **halt**.)

Table 4-2 Syntax for SRM Console Commands

Option	Attribute or Action
Length	**Up to 255 characters, not including the terminating carriage return or any characters deleted as the command is entered. A command longer than 80 characters and without the backslash character (see Table 4-3) causes display of an error message.**
Case	Upper- or lowercase characters can be used for input. Characters are displayed in the case in which they are entered.
Abbreviation	**Only by dropping characters from the end of words. You must enter the minimum number of characters to identify the keyword unambiguously. Abbreviation of environment variables is allowed with the show command.**
Options	You can use command options, to modify the environment, after the command keyword or after any symbol or number in the command. See individual command descriptions for examples.

Table 4-2 Syntax for SRM Console Commands (Continued)

Option	Attribute or Action
Numbers	**Most numbers in SRM Console commands are in decimal notation. Two exceptions, both of which use hexadecimal notation, are addresses and numbers used in the deposit command. The default radix can be overridden by inserting %d before the numbers you want to express in decimal, %o before octal, or %x before hexadecimal. Register names (for example, R0) are not considered numbers and use decimal notation.**
No characters	A command line with no characters is a null command. The SRM Console program takes no action and does not issue an error message; it returns the SRM Console prompt. The SRM Console supports command line recall and editing.
Spaces or Tabs	**Multiple adjacent spaces and tabs are compressed and treated as a single space. The SRM Console program ignores leading and trailing spaces.**

Table 4-3 Special Characters for SRM Console

Character	Function
Return or Enter	**Terminates a command line. No action is taken on a command until it is terminated. If no characters are entered and this key is pressed, the SRM Console just redisplays the prompt.**
Backslash (\)	Continues a command on the next line. Must be the last character on the line to be continued.
Delete	**Deletes the previous character.**
Help	By itself, displays first-level help. When pressed after part of a command, displays options available.
Ctrl/A or F14	**Toggles between insert and overstrike modes. The default is insert mode.**
Ctrl/B or up-arrow	Recalls previous command or commands. The last 16 commands are stored in the recall buffer.
Ctrl/C or Ctrl/P	**Terminates the process that is running. Clears Ctrl/S; resumes output suspended by Ctrl/O. When entered as part of a command line, deletes the current line. Ctrl/C has no effect as part of a binary data stream.**

Table 4-3 Special Characters for SRM Console (Continued)

Character	Function
Ctrl/D or left-arrow	Moves the cursor left one position.
Ctrl/E	**Moves the cursor to the end of the line.**
Ctrl/F or right-arrow	Moves the cursor right one position.
Ctrl/H or Backspace or F12	**Moves the cursor to the beginning of the line.**
Ctrl/J	Deletes the previous word.
Ctrl/O	**Stops output to the SRM Console terminal for the current command. Toggles between enable and disable. The output can be reenabled by other means as well: when the SRM Console prompts for a command, issues an error message, or enters program mode, or when Ctrl/P is entered.**
Ctrl/Q	Resumes output to the SRM Console terminal that was suspended by Ctrl/S.
Ctrl/R	**Redisplays the current line. Deleted characters are omitted. This command is useful for hardcopy terminals.**
Ctrl/S	Suspends output to the SRM Console terminal until Ctrl/Q is entered. Cleared by Ctrl/C.
Ctrl/U	**Deletes the current line.**
*	Wildcarding for commands such as **show**.
" "	**Double quotes enable you to denote a string for environment variable assignment.**
#	Specifies that all text between it and the end of the line is a comment. Control characters are not considered part of a comment.

4.2.3 Displaying the System Configurations

Several commands are used to display the system configuration:

- show config
- show cpu
- show device
- show memory
- show pal
- show power
- show version

show config

The **show config** command displays a list of devices found on the system interconnect and I/O buses. This is the configuration at the most recent initialization. The syntax is:

show config

Example 4-1 Show Config Command

```
P00>>>show config

                    AlphaPC 264DP 666 MHz

SRM Console:    A5.5-82
PALcode:        OpenVMS PALcode V1.61-49, Tru64 UNIX PALcode V1.54-50

Processors
CPU 0           Alpha 21264-A 666 MHz   SROM Revision: V1.9.2
                Bcache size: 4 MB
CPU 1           Alpha 21264-A 666 MHz   SROM Revision: V1.9.2
                Bcache size: 4 MB

Core Logic
Cchip           DECchip 21272-CA Rev 2.1
Dchip           DECchip 21272-DA Rev 2.0
Pchip 0         DECchip 21272-EA Rev 2.2
Pchip 1         DECchip 21272-EA Rev 2.2

TIG             Rev 4.14
Arbiter         Rev 2.8 (0x1)

MEMORY

Array #      Size     Base Addr
-------    ----------  ---------
  0        512 MB    000000000

Total Bad Pages = 0
Total Good Memory = 512 MBytes

PCI Hose 00
    Bus 00  Slot 05/0: Cypress 82C693 Bridge to Bus 1, ISA
    Bus 00  Slot 05/1: Cypress 82C693 IDE

                            dqa0.0.0.105.0        WDC AC29100D
    Bus 00  Slot 05/2: Cypress 82C693 IDE

                            dqb.0.1.205.0
    Bus 00  Slot 05/3: Cypress 82C693 USB

    Bus 00  Slot 06: Adaptec AIC-7891
                             pka0.7.0.6.0          SCSI Bus ID7
```

```
    Bus 00  Slot 07: 00093D3D/01003D3D

PCI Hose 01
    Bus 00  Slot 08: 920010B7/100010B7


ISA
Slot   Device  Name          Type        Enabled  BaseAddr  IRQ
DMA
0
       0       MOUSE         Embedded     Yes     60        12
       1       KBD           Embedded     Yes     60        1
       2       COM1          Embedded     Yes     3f8       4
       3       COM2          Embedded     Yes     2f8       3
       4       LPT1          Embedded     Yes     3bc       7
       5       FLOPPY        Embedded     Yes     3f0       6
2
```

show cpu

The **show cpu** command displays the status of each CPU. The syntax is:

show cpu

Example 4-2 Show CPU Command

```
P00>>>show cpu

Primary CPU:      00
Active CPUs:      00     01
Configured CPUs:  00     01
SROM Revision:    V1.9.2  V1.9.2
```

show device

The **show device** command displays status for devices and controllers in the system: SCSI and MSCP devices, the internal floppy drive, and the network. The syntax is:

show device *[controller_name]*

controller_name	The controller name or abbreviation. When abbreviations or wildcards are used, all controllers that match the type are displayed. If no name is given, the display is a list of all devices and controllers in the system.

Example 4-3 Show Device Command

```
P00>>>show device
dqa0.0.0.105.0      DQA0      WDC AC29 100D  J74OA30K
dva0.0.0.0.0        DVA0
pka0.7.0.6.0        PKA0      SCSI Bus ID 7
```

An example of a device name is **dka200.2.0.7.**1. Table 4-4 shows the interpretation of this device name.

Table 4-4 Device Naming Convention

Category	Description
dk Driver ID	**Two-letter designator of port or class driver:** **dk SCSI device** **fw FDDI device** **dq ATAPI CD-ROM** **mk SCSI tape** **dr RAID set device** **mu DSSI tape** **du DSSI disk** **pk SCSI port**
a Storage adapter ID	One-letter designator of storage adapter (a, b, c...).
200 Device unit number	**Unique number (MSCP unit number). SCSI unit numbers are forced to 100 X node ID.**
2 Bus node number	Bus node ID.
0 Channel number	**Used for multi-channel devices.**
7 Logical slot number	Corresponds to PCI slot number.
1 Hose number	**0 — PCI 0 1 — PCI 1**

Table 4-5 PCI Address Assignments

Slot	PCI 0	PCI 1	ISA
5	**ISA bridge (on board)**		**ISA device**
6	Adaptec SCSI (on board)		
7	**PCI device**	**PCI device**	
8	PCI device	PCI device	
9	**PCI device**	**PCI device**	

show memory

The **show memory** command displays information about each memory bank: slot number, size in megabytes, and the starting address. The syntax is:

show memory

Example 4-4 Show Memory Command

```
P00>>>show memory

Array #      Size       Base Addr
-------      ----------  ---------
  0          512 MB     000000000

Total Bad Pages = 0
Total Good Memory = 512 MBytes
```

show pal

The **show pal** command displays the versions of PALcode. PALcode is the Privileged Architecture Library (PAL) code, written to support Alpha processors. It implements architecturally defined processor behavior. The syntax is:

show pal

Example 4-5 Show PAL Command

```
P00>>>show pal
pal                OpenVMS PALcode V1.61-49, Tru64 UNIX PALcode
V1.54-50
```

show power

The **show power** command displays status information about the power supplies, system fans, CPU fans, and temperature. This command is useful for displaying the error state of a system that shuts down because of a fan, temperature, or power supply failure. If the system can be restarted, use this command. The syntax is:

show power

Example 4-6 Show Power Command

```
P00>>> show power

                    Status
Power Supply        good
CPU 0 Power         good
CPU 1 Power         good
```

```
CPU 0 Temp      good
CPU 1 Temp      good
CPU 0 Fans      good
CPU 1 Fans      good
```

show version

The **show version** command displays the version of the SRM Console program that is installed on the system. The syntax is:

show version

Example 4-7 Show Version Command

```
P00>>>show version
version             A5.5-82 Dec 29 1999 09:08:45
```

4.2.4 Booting the Operating System

The **boot** command is used to boot the operating system. The **boot** command initializes the processor, loads a program image from the specified boot device, and transfers control to that image. The syntax is:

boot [-file *filename***] [-flags [***value***]] [-halt] [-protocols** *enet_protocol***] [***boot_de***v]**

Table 4-6 Boot Command Options

Option	Description
-file *filename*	**The boot file.**
-flags [*value*]	Specifies additional information to the loaded image or operating system.This qualifier overrides the setting of the **boot_osflags** environment variable. See the **boot_osflags** environment variable on page 4-17 for a list of settings and their meanings.
-halt	**Forces the bootstrap operation to halt and invokes the SRM Console program once the bootstrap image is loaded and page tables and other data structures are set up. Operator console device drivers are not shut down. Transfer control to the image by entering the continue command.**
-protocols *enet_protocol*	Either **mop** or **boot**p (default). This qualifier overrides the setting of the **ew*0_protocols** environment variable (see Table 4-12).
boot_dev	**A device path or list of devices from which the SRM Console program attempts to boot, or a saved boot specification in the form of an environment variable. This qualifier overrides the setting of the bootdef_dev environment variable (see page 4-17). Use the** bootdef_dev **environment variable to define the default boot device string.**

Note: The operator console is the monitor, keyboard, and mouse. This hardware is used to enter SRM Console commands into the system.

Example 4-8 Boot Command

```
P00>>> b dka200

(boot dka200.2.0.7.1 -flags 0,0)
block 0 of dka200.2.0.7.1 is a valid boot block
reading 893 blocks from dka200.2.0.7.1
bootstrap code read in
base = 1fa000, image_start = 0, image_bytes = 6fa00
initializing HWRPB at 2000
initializing page table at 1fff0000
initializing machine state
setting affinity to the primary CPU
jumping to bootstrap code
```

4.2.5 Updating Firmware

The **lfu** command is used to update firmware from the SRM Console prompt. The **lfu** command starts the Loadable Firmware Update (LFU) Utility. The syntax is:

lfu

Note: *If the system has been shut down from a booted program (most commonly, the operating system) or in some other way halted back to the SRM Console, the system must be reset before running LFU.*

Example 4-9 Lfu Command

```
P00>>>lfu

Checking dva0 for the option firmware files. . .

Option firmware files were not found on CD or floppy.
If you want to load the options firmware,
please enter the device on which the files are located(ewa0),
or just hit <return> to proceed with a standard console update: dva0
Please enter the name of the options firmware files list, or
Hit <return> to use the default filename (pc264fw.txt) :
Copying pc264fw.txt from dva0. . .
Copying pc264srm.rom from dva0. . .


                ***** Loadable Firmware Update Utility *****
--------------------------------------------------------------------------------
 Function    Description
--------------------------------------------------------------------------------

 Display      Displays the system's configuration table.
 Exit          Done exit LFU (reset).
 List          Lists the device, revision, firmware name, and update
revision.
 Readme      Lists important release information.
 Update       Replaces current firmware with loadable data image.
 Verify        Compares loadable and hardware images.
 ? or Help    Scrolls this function table.
--------------------------------------------------------------------------------


UPD> update srm

Confirm update on:
srm
[Y/(N)]y
WARNING: updates may take several minutes to complete for each device.
```

```
                    DO NOT ABORT!

srm         Updating to 5.5-82...  Verifying 5.5-82...  PASSED.

UPD>
```

4.2.6 Using Environment Variables

Environment variables pass configuration information between the SRM Console and the operating system. Their settings determine how the system powers up, boots the operating system, and operates. You issue an **init** command (see 4.2.8 on page -24 for more details) to activate a new environment variable.

Example 4-10 Set *envar* and Show *envar* Commands

```
P00>>> show console
console                 graphics
P00>>> set console serial
P00>>> show console
console                 serial
P00>>> init
```

Environment variables are set or changed with the **set *envar*** command and set to default values with the **set -default *envar*** command. Their values are viewed with the **show *envar*** command. User-defined nonvolatile environment variables are created with the **edit** (see 4.2.8 on page -24 for further information) command.

set *envar*

The **set** command sets or modifies the value of an environment variable. It can also be used to create a new environment variable if the name used is unique. Environment variables are used to pass configuration information between the SRM Console and the operating system. The setting of these variables determines how the system powers up, boots the operating system, and operates. The syntax is:

set [-default] *envar value*

Table 4-7 Set *Envar* Options

Option	Description
-default	**Restores an environment variable to its default setting.**
envar	The name of the environment variable to be modified.
value	**The new value of the environment variable.**

Whenever you modify the value of any of the following environment variables, the new value takes effect only after you reset the system by pressing the **Reset** button or issuing the **initialize** command:

console

kbd_hardware_type

language

os_type

show *envar*

The **show *envar*** command displays the current value (or setting) of an environment variable. The syntax is:

show *envar*

envar	The name of the environment variable to be displayed. The wildcard * displays all environment variables.

Example 4-11 Using show *envar*

```
P00>>>show os_type
os_type
P00>>>unix
```

4.2.7 Environment Variable Summary

Environment variables pass configuration information between the SRM Console and the operating system. Their settings determine how the system powers up, boots the operating system, and operates. Environment variables are set or changed with the **set** *envar* command and returned to their default values with the **clear** *envar* command. Their values are viewed with the **show** *envar* command.

Table 4-8 lists the environment variables. Detailed descriptions follow. The environment variables are specific to the SRM Console.

Table 4-8 Environment Variable Summary

Environment Variable	Function
auto_action	**Specifies the SRM Console's action at power-up, a failure, or a reset.**
bootdef_dev	Specifies the default boot device string.
boot_osflags	**Specifies the default operating system boot flags.**
com*_baud	Changes the default baud rate of the COM1 or COM2 serial port.
console	**Specifies the device on which power-up output is displayed (serial terminal or graphics monitor).**
cpu_enabled	Enables or disables a specific secondary CPU.
ew*0_mode	**Specifies the connection type of the default Ethernet controller.**
ew*0_protocols	Specifies network protocols for booting over the Ethernet controller.
kbd_hardware_ type	**Specifies the default operator console keyboard type.**
language	Specifies the operator console keyboard layout.
os_type	**Specifies the operating system. Valid entry is: unix.**
password	A password stored in the NVRAM used to secure the operator console.
pci_parity	**Disables or enables parity checking on the PCI bus.**
pk*0_fast	Enables fast SCSI mode.
pk*0_host_id	**Specifies the default value for a controller host bus node ID.**
pk*0_soft_term	Enables or disables SCSI terminators on systems that use the QLogic ISP1040 SCSI controller.
tt_allow_login	**Enables or disables login to the SRM Console firmware on other operator console ports.**

auto_action

Specifies the action the SRM Console takes any time the system powers up, fails, or resets. When the setting involves autoboot, the system boots from the default boot device specified by the value of the **bootdef_dev** environment variable. The syntax is:

set auto_action *value*

The options for *value* are shown in Table 4-9.

Table 4-9 Auto_Action Options

Option	Description
halt	**The system remains in SRM Console mode after power-up or a system crash.**
boot	The system boots automatically when it is turned on and halts after a system failure.
restart	**The system boots automatically when it is turned on or after it fails.**

Note: If a halt assertion exists, the SRM Console ignores the auto_action setting and halts at the SRM Console.

bootdef_dev

The **bootdef_dev** environment variable specifies one or more devices for booting the operating system. When more than one device is listed, the system searches in the order listed and boots from the first device with operating system software. The syntax is:

set bootdef_dev *boot_device*

boot_device	The name of the device on which the system software has been loaded. To specify more than one device, separate the names with commas. Enter the command **show bootdef_dev** to display the current default boot device. Enter the command show device for a list of all devices in the system.

boot_osflags

The **boot_osflags** environment variable passes information to the **boot** command. That information is dependent on the operating system to be booted. The syntax is:

set boot_osflags *flags_value*

The options for *flags_value* are shown in Table 4-10.

Table 4-10 Boot_Osflags Options

Option	Description
root=/dev/sda5	**Set the root filesystem to the 5th partition of the first SCSI disk.**
root=/dev/hda2	Set the root filesystem to the 2nd partition of the first IDE disk.
1	**Use config number 1 from the /etc/aboot.conf file**

com*_baud

The default baud rate for the system is 9600. With the **com*_baud** environment variable, you can set the baud rate to match that of the device connected to the port.

You will be asked to confirm the change, as shown here:

```
P00>>> set com1_baud 19200
Embedded Remote Console only supports 9600 baud. Continue?
(Y/[N]) n
bad value - com1_baud not modified
P00>>>
```

The syntax is:

set com*_baud *baud_value*

baud_value	The new baud rate. A list of possible values is displayed by attempting to set this environment variable to an unacceptable value (for example, **set com2_baud xxx**).

console

The operator console terminal can be either a graphics monitor or a serial terminal. The **console** environment variable specifies which is used. The syntax is:

set console *output_device*

The options for *output_device* are:

graphics (default)	The operator console terminal is a graphics monitor or a device connected to the VGA.
serial	The operator console terminal is the device connected to the COM1 port.

Whenever you change the value of **consol**e, you must reset the system by pressing the **Reset** button or issuing the **initialize** command.

cpu_enabled

Sets a bit mask that enables or disables specific CPUs on a multiprocessor system. Disabled CPUs are prevented from running the SRM Console or the operating system. Bit 0 of the mask corresponds to CPU 0, and bit 1 to CPU 1. A zero in the bit mask prevents the corresponding CPU from running; a one allows it to run. The bit mask is expressed as a hexadecimal value.

Note: *The primary CPU cannot be disabled.*

The syntax is:

set cpu_enabled *hex_digit*

The options for *hex_digit* are 0 and 1 (hexadecimal).

ew*0_mode

Sets an Ethernet controller to run an Ethernet network. The default value is **auto-sense**. For the fast setting, the device defaults to fast.

The syntax is:

set ew*0_mode *value*

The options for *value* are shown in Table 4-11.

Table 4-11 ew*0_mode Options

Option	Description
aui	**Device type is AUI.**
auto-sense	Device type is sensed by the SRM Console.
twisted-pair	**Device type is 10BaseT (twisted pair).**
fast duplex, twisted-pair	Device type is duplex 10BaseT
fast	**Device type is fast 100Base TX**
fast FD	Device type is fast full duplex 100Base TX
BNC	**Device type is BNC**
auto-negotiate	DE500-BA

ew*0_protocols

Enables network protocols for booting and other functions. The syntax is:

set ew*0_protocols *protocol_value*

The options for *protocol_value* are show in Table 4-11.

Table 4-12 ew*0_protocols Options

Option	Description
mop	**Sets the network protocol to mop (Maintenance Operations Protocol), the setting typically used with the Linux operating system.**
bootp (default)	Sets the network protocol to bootp, the setting typically used with the Linux operating system.
bootp, mop	**When both are listed, the system attempts to use the mop protocol first, regardless of which is listed first. If not successful, it then attempts the bootp protocol.**

kbd_hardware_type

Used only on systems with the language variant 3C (Français), this environment variable sets the keyboard hardware type as either PCXAL or LK411 and enables the system to interpret the terminal keyboard layout correctly.

Whenever you change the value of **kbd_hardware_typ**e, you must reset the system by pressing the **Reset** button or issuing the **initialize** command.

The syntax is:

set kbd_hardware_type *keyboard_type*

The options for *keyboard_type* are:

pcxal (default)	Selects the default keyboard hardware type.
lk411	Selects the LK411 keyboard layout for use with language variant 3C (Français).

language

Specifies the keyboard layout, which is language dependent. The setting of the **language** environment variable must match the language of the keyboard variant.

Whenever you change the value of **languag**e, you must reset the system by pressing the **Reset** button or issuing the **initialize** command.

The syntax is:

set language *language_code*

The options for *language_code* are show in Table 4-13.

Table 4-13 Language Options

Option	Description
0	**No language (cryptic)**
30	Dansk (Danish)
32	**Deutsch (German)**
34	Deutsch (Schweiz) (Swiss)
36	**English (American)**
38	English (British / Irish)
3A	**Español (Spanish)**
3C	Français (French)
3E	**Français (Canadian)**
40	Français (Suisse Romande)
42	**Italiano (Italian)**
44	Nederlands (Netherlands)
46	**Norsk (Norwegian)**
48	Portuguese (Portuguese)
4A	**Suomi (Finnish)**
4C	Svenska (Swedish)
4E	**Belgisch-Nederlands (Dutch)**

os_type

The **os_type** environment variable specifies the default operating system. This variable is set at the factory to the setting for the operating system purchased. Use this command to change the factory default setting.

Whenever you change the value of **os_typ**e, you must reset the system by pressing the **Reset** button or issuing the **initialize** command.

The syntax is:

set os_type *os_type*

The options for *os_type* are:

unix	Linux is the default operating system, and the SRM firmware is started during power-up or reset.

password

Sets or clears the SRM Console password stored in Non-Volatile RAM (NVRAM).

The syntax is:

set password

The password is not an argument to the **set password** command; the SRM Console prompts the user for the string, which must be between 15 and 30 characters.

pci_parity

Disables or enables parity checking on the PCI bus.

Some PCI devices do not implement PCI parity checking, and some have a parity-generating scheme in which the parity is sometimes incorrect or is not fully compliant with the PCI specification. A side effect of this aberrant behavior is that superfluous PCI parity errors are reported by the host PCI bridge. In such cases, the device can be used as long as parity is not checked; disabling PCI parity checking prevents false parity errors that can cause system problems.

The syntax is:

set pci_parity *value*

The options for *value* are:

on (default)	Enables PCI parity checking.
off	Disables PCI parity checking.

pk*0_fast

Enables fast SCSI to perform in either standard or fast mode. If the system has at least one fast SCSI device, set the default controller speed to fast SCSI (1). Devices on a controller that connects to both standard and fast SCSI devices will perform at the appropriate rate for the device. If the system has no fast SCSI devices, set the default controller speed to standard SCSI (0). If a fast SCSI device is on a controller set to standard, it will perform in standard mode.

The syntax is:

set pk*0_fast *scsi_speed*

The options for *scsi_speed* are:

0	The controller is in standard SCSI mode.
1 (default)	The controller is in fast SCSI mode.

pk*0_host_id

Sets the controller host bus node ID to a value between 0 and 7.

Each SCSI bus in the system requires a controller. Buses can theoretically support up to eight devices; however, the eighth device must always be a controller. Each device on the bus, including the controller, must have a unique ID, which is a number between 0 and 7. This is the bus node ID number.

On each bus, the default bus node ID for the controller is set to 7. You do not need to change the controller bus node ID unless you place two or more controllers on the same bus.

To list the controllers on your system, enter the command **show device** (see page 4-8). SCSI devices begin with the letters "pk" (for example, pka0). The third letter is the adapter ID for the controller. When entering the command **set pk*0_host_i**d, replace the asterisk with the adapter ID letter.

The syntax is:

set pk*_host_id *scsi_node_id*

The value for *scsi_node_id* is the bus node ID, a number from 0 to 7.

pk*0_soft_term

Enables or disables SCSI terminators. This command applies to systems that use the QLogic ISP1040 SCSI controller.

The QLogic ISP1040 SCSI controller implements the 16-bit wide SCSI bus. The QLogic module has two terminators, one for the low eight bits and one for the high eight bits.

The syntax is:

set pk*0_soft_term *value*

The options for *value* are shown in Table 4-14.

Table 4-14 pk*0_soft_term Options

Option	Description
off	**Disables termination of all 16 bits.**
low (default)	Enables low eight bits and disables high eight bits.
high	**Enables high eight bits and disables low eight bits.**
on	Enables all 16 bits.
diff	**Places the bus in differential mode.**

tt_allow_login

Enables or disables login to the SRM Console firmware on alternate operator console ports. If the environment variable **console** (see page 4-18) is set to serial, the primary operator console device is the terminal connected through the COM1 port. The command **set tt_allow_login 1** enables logins through either the COM2 port or a graphics monitor.

The syntax is:

set tt_allow_login *value*

The options for *value* are:

0	Disables login through the COM2 port or a graphics monitor.
1 (default)	Enables login through the COM2 port or a graphics monitor.

4.2.8 Finding Help

The **help** command displays basic information about SRM Console commands.

Example 4-12 Help Command

```
P00>>> help set
NAME
    set
FUNCTION
    Set an option or modify the value of an environment
    variable.
```

```
SYNOPSIS
    set <option> <value> or <envar> [-] <value>
    where
    <option>={host,mode}
    where
    <envar>={auto_action,bootdef_dev,boot_osflags,...}
    [-default]
```

The **help** command displays basic information about the use of SRM Console commands when the system is in SRM Console mode. The syntax is:

help *[command* **. . .** *]*

command . . .	Command or topic for which help is requested. The options are:

none	Displays the complete list of commands for which you can receive help.
command_name	Displays information about the SRM Console command.
argument_string (such as "sh")	Displays information about all commands that begin with that string.

4.3 Alpha Diagnostics

The Alpha Diagnostics firmware is used internally by Alpha Processor, Inc. for diagnostic purposes.

Native mode diagnostics depends on various system components to be functioning correctly. When Reset PAL code determines that the UP2000 is capable of supporting the higher level environment, it fetches this image from the firmware and transfers control to it.

An Alpha Slot B Module using an 21264 processor implements a serial communications link directly connected to the processor. This link, called the Debug Port, can be used for reporting and interacting in the earliest stages of system initialization, after execution passes from PAL mode. It is accessed through J25 on the UP2000 Motherboard. See Chapter 2 for more information.

The Alpha Diagnostics includes the following tests:

- Interrupt handling—Raise interrupts with a known response
- UP2000 Motherboard components—chipset, Flash ROM integrity, on-board devices
- Memory—stress test
- ISA cards
- PCI bus—Initialization, stressing and interrupts
- SM timer support and EEPROMs
- FDD and IDE disks—DMA

If the Alpha Diagnostics detects a working keyboard and video console, it displays a graphical interface containing a menu of diagnostics. This is the Console interface to the Alpha Diagnostics. If the Alpha Diagnostics does not detect a video console, the Alpha Diagnostics uses the Debug Port interface.

4.4 Installing the Linux Operating System

The firmware initializes the UP2000 system and, via the Alpha SRM Console, enables you to install and boot various operating systems.

As noted at the beginning of Chapter 4, “Software Support” this system supports many OS distributions and vendors. Consult our Web site for a complete current list.

Examples of installing commercially available Linux distributions are given in this section. Typical requirements and procedures for Red Hat Linux Versions 6.1 or SuSE Version 6.3 have been selected.

4.4.1 Requirements

The appropriate Linux operating system distribution CD-ROM disk is required to install the OS. In this example, the Red Hat Version 6.1 or SuSE Version 6.3 CD-ROM disk is required.

4.4.2 Before Installing Linux

Setting Environmental Variables

From the SRM Console prompt, P00>>>, check the device numbers for disk, diskette and CD-ROM drives:

Type **show device**

The SRM Console environmental variables identified in Table 4-15 are set.

Table 4-15 SRM Console Variables

Variable	Setting
bootdef_dev	**Sets default boot device** **Example: DQA0 or DKA0**
boot_osflags	Information passed to Linux kernel via boot Example: root=/dev/hda
boot_file	**Sets file to use as the kernel on the default boot device** **Example for Red Hat 6.1: kernels/generic.gz**

From the SRM Console prompt, P00>>>, the procedures are as follows:

- To set the default boot device to the CD-ROM:
  Type **set bootdef_dev dqa0**
- To set the default boot file to the kernel on the CD, choose the command that corresponds to your version of Linux:
  - For Red Hat 6.1:
    Type **set boot_file "kernels/generic.gz"**
  - For SuSE 6.3:
    Type **set boot_file**
- To set the default flags to point to the (currently non-existent) root partition on the hard drive:
  Type **set boot_osflags "root=/dev/sda1"**
  Note: *Setting the flags to point to sda1 (Linux terminology for SCSI Disk A Partition 1) assumes that you will create and set the first partition during the installation to be the "root" partition. If you plan to use another name for your "root" partition, set this variable to that name.*
- To check the environmental setting parameters you have chosen:
  Type **show boot***

4.4.3 Starting the Linux OS Installation

To start the Linux OS, perform the following:

1. Insert the Red Hat Version 6.1 or SuSE Version 6.3 Alpha CDROM into the drive.
2. From the SRM Console prompt (P00>>>) for Red Hat Version 6.1:
   Type **boot -flags "root=/dev/hda"**
   or

From the SRM Console prompt (P00>>>) for SuSE Version 6.3:
Type **boot -flags 0**

3. Follow the instructions for your distribution.

Background Information

To the SRM Console, the IDE CD-ROM drive is called DQA0. To the Linux kernel, it is called "/dev/hda".

For this initial installation boot, you use the root directory of the CD-ROM. In this case, by choosing the boot parameter **-flags "root=/dev/hda"** to begin, you bypass the environmental setting made in section 4.4.2.

Device names are different if you are using SCSI adapters or IDE disks. To the Linux kernel:

- SCSI CD-ROM names are "/dev/scd***x***", where ***x*** is the device number
- SCSI hard disk names are "/dev/sd***x***"
- IDE devices are "/dev/hd***x***"

Other Boot Options

Refer to your Linux documentation in order to consider other boot options.

To boot, the Linux kernel must be on an SRM-supported device.

To boot from a diskette, set the environmental variable **bootdef_dev** to the diskette drive (dva0).

To boot from a hard disk drive (dka0), set the environmental variable **boot_file** to the directory and filename of the kernel.

Instructions for creating a SRM bootable diskette or hard disk are available on-line at:

http://www.alpha-processor.com/support/srm.html

or

http://www.alphalinux.org/faq/SRM.html

Post Installation Check

At this point, Linux is installed on the hard disk. If Partition 1 is the root directory, then the environmental variable **boot_osflags** is correct. If your root partition is different, take this opportunity to set the variable to the corresponding name.

Chapter 5 Troubleshooting

This chapter discusses troubleshooting aspects for both hardware and software components during the UP2000 system startup.

Topics covered include:

- Video presence checklist
- Status LEDs
- POST codes
- Beep codes
- Error recovery procedures

5.1 Hardware Startup

5.1.1 Video Presence Checklist

Use the following steps to diagnose and fix video problems:

1. Check the AC power cord connection to the AC outlet.
2. Ensure that the monitor is switched on.
3. Check the voltage setting on the chassis power supply (115 VAC in the U.S.).
4. Check that the CPU speed configuration jumpers (J27, J29, J34) are selected properly, either 667 MHz or 750 MHz.

WARNING: ***Always take appropriate electrostatic discharge safety measures when handling boards or modules.***

5. Check that the Alpha Slot B Module fans are connected and spinning.
6. Turn the system power OFF.
7. Reseat the video card and ensure that it is connected to the monitor.
8. Reseat the DIMMs.
9. Replace the DIMMs.

5.1.2 LED Status Indicators

Eight square LED indicators provide diagnostic information about a UP2000 system including the status of certain Alpha Slot B Module

functions.

The LEDs are mounted on the lower edge of the motherboard *between* the Secondary Alpha Slot B Module and the Internal I/O connector area (J40 through J47). Their orientation is shown in Figure 5-1.

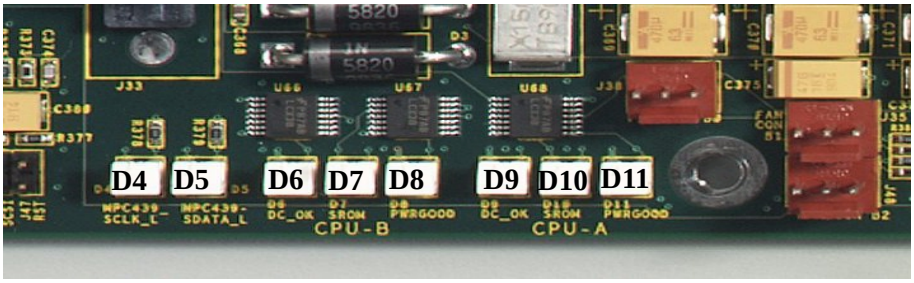

Figure 5-1 LED Status Indicators

Use Table 5-1 to interpret the LED status information

Table 5-1 LED Status Indicators

LED from photo	Function	Comment
D4	**SCLK_L**	**ON while TIGPAL sends clock to Clock Generator**
D5	SDATA_L	ON while TIGPAL sends data to Clock Generator
Secondary Alpha Slot B Module CPU:		
D6	DC_OK	ON when reset FPGA senses the DC_OK signal
D7	**SROM**	**Fast flash (LED appears ON, but dim) while SROM is being loaded**
D8	PWRGOOD	ON when reset FPGA senses the Core-Power Good signal
Primary Alpha Slot B Module CPU:		
D9	DC_OK	ON when reset FPGA senses the DC_OK signal
D10	**SROM**	**Fast flash (LED appears ON, but dim) while SROM is being loaded**
D11	PWRGOOD	ON when reset FPGA senses the Core-Power Good signal

5.1.3 POST (Power On Self Test) Codes

Utilizing an ISA-based POST card module, you can monitor the sequential steps as the system is initialized from the SROM. Each post code, its source and a description of its message is listed in Table 5-2.

Table 5-2 POST Codes Information

Source	POST Code (hex)	Message
SROM	**20**	**Firmware initialization complete**
	01	CPU speed detected; initialize Southbridge
	02	**Southbridge ready**
	03	Initialize L2 cache
	05	**Start sweep of L2 cache and memory**
	06	L2 cache and memory ready
	14	**Load system code**
	15	Loading uncompressed firmware into memory
	16	**Loading compressed firmware into memory**
	17	Jump to loaded firmware
	3F	**Fatal error. Second code identifies source of error 06 = no memory found or bad memory**

5.1.4 Beep Codes

The UP2000 system delivers several audible troubleshooting messages which are referred to as beep codes. They are described in Table 5-3.

Table 5-3 Beep Codes Information

Beep Code	Message Description
4	**No valid header found in ROM; loading entire ROM**
6	Memory error detected

5.2 Error Recovery Procedures

5.2.1 Error Situations

The Alpha Diagnostics utility provides the emergency recovery

mechanism when the primary firmware image contained in flash memory is corrupted. When flash memory is corrupted, and no image can be loaded safely from the flash ROM, you can run Alpha Diagnostics and boot another image from a diskette that is capable of reprogramming the flash ROM.

For more information, see the FAQs on the Alpha Processor, Inc. web site:

http://www.alpha-processor.com

5.2.2 Error Recovery Jumper Settings

The UP2000 Motherboard has configuration jumpers located at J29 which have selectable settings. (See Figure 5-2.) You use these jumpers to restore firmware factory defaults or recover default firmware. By default, a jumper is not installed on pins 0 and 1 (bit 0) of J29 on the UP2000 Motherboard (the OFF position), and Alpha Diagnostics is not enabled.

Configuration of diagnostics and flash recovery is managed through settings of pins 1–2, 6–7, and 14–15 of J29. Table 5-4 shows the possible configuration settings for diagnostics and flash recovery of the UP2000. The default function is Normal Boot Sequence, with no shunts installed on pins 1–2, 6–7, and 14–15 of J29.

Note: *All remaining jumpers on J29 must remain in the default positions. These jumpers control other functions.*

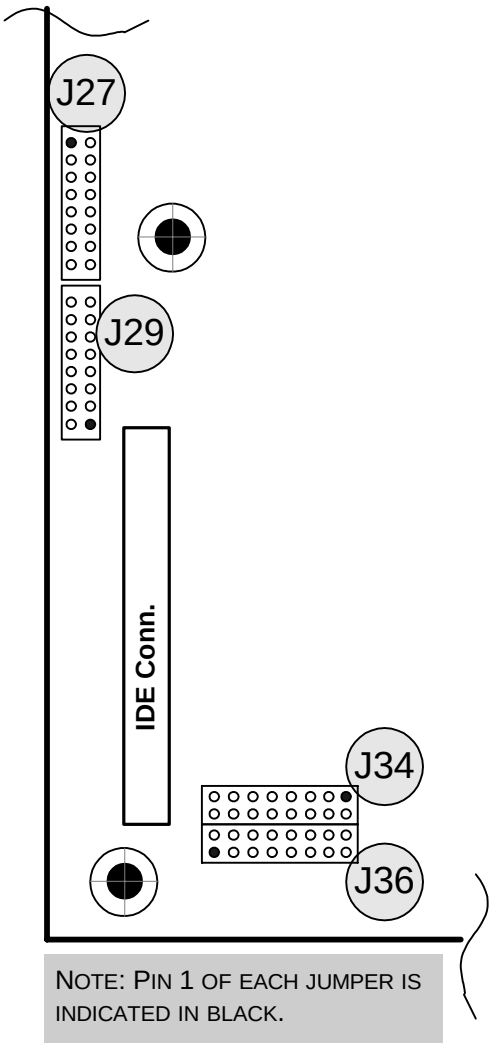

Figure 5-2 J29 Configuration Jumper Block

To perform error recoveries, use the appropriate configuration jumper settings listed in Table 5-4.

Table 5-4 Diagnostics and Flash Recovery Configuration Settings (J29)

Function	Install Jumper on Pins:		
	0/1	6/7	14/15
Normal Boot Sequence (using firmware)—Default	**0**	**0**	**0**
Restore Factory Defaults (Bypass password check)	0	0	1
Normal Recovery (using firmware)	**1**	**0**	**0**
Alpha Diagnostics Recovery (using firmware)	1	0	1
Interactive Alpha Diagnostics (using operator console)	**1**	**1**	**0**
Interactive Alpha Diagnostics (using Debug Port J42/J43)	1	1	1
Interactive Reset PAL Code (using Debug Port J42/J43)	**0**	**1**	**0**

Notes: ***1. 0 = No shunt installed. 1 = Shunt installed.***

2. The Debug Ports (primary Alpha Slot B Module port J43 and secondary Alpha Slot B Module port J42) default to 9600 baud. When using the Interactive Reset PAL code function, type an uppercase U to set the interactive baud rate higher.

5.2.3 Error Recovery Procedure

To clear the errors noted in Section 5.2.1, take the following steps:

1. Power off the system.
2. Change the jumper shunts according to the error to be cleared.
3. Start the system.
4. Enter the proper parameters in SRM Console, load the Operating System.
5. Power off the system.
6. Restore the jumper shunts to their default positions.
7. Start the system.

Appendix A

Support,

Products and

Documentation

A.1 Customer Support

API provides assistance for their products on their web page at www.alpha-processor.com.

Alpha Original Equipment Manufacturers (OEMs) provide the following web page resources for customer support:

URL	Description
http://www.compaq.com	**Contains links for the 21272 chipset.**
http://www.samsungsemi.com	Contains links for the 21264 CPU.

A.2 Supporting Products

API maintains a Hardware Compatibility List on their web site for components and accessories that are not included with the UP2000. Compatibility for items such as memory, power supplies, and enclosure are listed.

Point your browser to www.alpha-processor.com and check the Product Information list for Peripherals.

A.3 Alpha Products

API maintains information about other Alpha products on their web site. Point your browser to www.alpha-processor.com and check the Product Information list for Alpha products.

A.4 Related Publications

A.4.1 Alpha Processor, Inc. Publications

The UP2000 suite of publications available from Alpha Processor, Inc. include the following:

UP2000 Quick Start Installation Guide

The *UP2000 Quick Start Installation Guide*, part number 51-0030-1A, is an illustrated step-by-step report focused on setting up and configuring a UP2000 system.

UP2000 Technical Reference Manual

The *UP2000 Technical Reference Manual*, part number 51-0029-3A, is an illustrated manual describing the Alpha Processor, Inc. UP2000 product, including the UP2000 Motherboard used with the Alpha Slot B Module.

A.4.2 Third-Party Publications

The following documents are useful for understanding and navigating the UP2000 system and its user interface.

Linux

Linux for Dummies, 2nd Edition, Jon "maddog" Hall, IDG Books Worldwide, Inc., Foster City, CA 94404

Alpha Architecture

Alpha Architecture Reference Manual, Third Edition, Digital Press, order # EQ-W938E-DP.

Alpha Architecture Handbook, Version 4, Compaq Computer Corporation, order # EC-QD2KC-TE, October, 1998.

Index

Numerics

A

B

C

D

E

R

S

T

U

V

Publication Evaluation Form

Please help us develop better manuals by sending us your comments on our publications. You can submit your comments on-line, send us E-mail, or print this evaluation form, complete it, and return it to:

Alpha Processor, Inc.
130C Baker Street Extension
Concord, MA 01742
Phone: 978.318.1100 FAX: 978.371.3177
E-mail: customer.support@alpha-processor.com

Access on-line form

Reset form

Contents and Organization

Publication Name:

I ◯ Agree ◯ Disagree that the information I need is in this publication.

I ◯ Agree ◯ Disagree that it is easy to locate information in this publication.

I ◯ Agree ◯ Disagree that I am confident of the accuracy of this publication.

What I liked least about this publication is:

What I like most about this publication is:

Information not provided in this publication that I would find useful is:

This publication would be easier to use if:

Other comments:

User Information (Optional)

Name:		Company:	
Street Address:		Phone:	
City, State, Zip:		E-mail:	