UP2000+
User Manual

Part Number: 51-0042-1C

Part #: **51-0042-1C**

Date: **1/12/01**

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

API NetWorks, 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. API NetWorks, 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 reserve the right to make changes at any time, without notice. API NetWorks, 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. API NetWorks, Inc. products are not authorized for use as critical components in life support devices or systems without API NetWorks, Inc. written approval. API NetWorks, Inc. assumes no liability whatsoever for claims associated with the sale or use (including the use of engineering samples) of API NetWorks, Inc. products except as provided in API NetWorks, Inc. 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 NetWorks, 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.

Revision History

Date	Rev	Description
6/22/00	**51-0042-1A**	***UP2000+ User Manual* first product release. This manual describes the API NetWorks, Inc. part number UP2000-B1.**
7/24/00	51-0042-1B	Add DDR Slot B support.
12/11/00	51-0042-1C	Configuration requirements corrected for jumper block J31.
12/28/00	51-0042-1C	Logo, URL, and company name updated.

Table of Contents

List of Figures

List of Tables

Preface

Overview

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

Task-oriented 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:

- Configure the memory subsystem
- Configure the output voltage for the CPU

Audience

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

Scope

This manual describes the features, configuration options, functional operation, troubleshooting analysis and user interface of the system and its firmware. It is a companion piece to the API NetWorks, Inc. UP2000+ document set that includes the *UP2000+ Quick Start Installation Guide* (51-0041) and the *UP2000+ Technical Reference Manual* (51-0039).

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 main memory guidelines, firmware configuration settings, and CPU configuration settings.

- Chapter 3, “Electrical 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 a Linux operating system, as well as displaying and configuring system parameters.
- Chapter 5, “Troubleshooting,” discusses solutions for hardware and software problems encountered during system startup. System upgrade and system error recovery techniques are provided.
- Pinouts for the non-standard connectors used in the UP2000+ are provided in Appendix A, “Connectors and Pinouts.” This appendix also contains a list of all other connectors used in the UP2000+, which are industry standard parts.
- Appendix B, “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.

Typographic Conventions

This manual uses the following type conventions:

- Variable information and document titles appear in *italic* type.
- Text that you type is shown in **`bold Courier font`**.
- Type that appears on a screen, such as an example of computer output, is shown in `Courier font`.
- Two key names joined with a forward slash are simultaneous keystrokes. Press down the first key while you type the second key, as in press Ctrl/S.

Acronyms

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

Acronym	Meaning
API	Application Program Interface
BIOS	Basic Input/Output System
CE	European Conforming
CPU	Central Processing Unit
cUL	Canadian Underwriters Laboratory
DBM	Debug Monitor
DDR	Double Data Rate
DIMM	Dual Inline Memory Module
DMA	Direct Memory Access
DRAM	Dynamic Random Access Memory
EIDE	Enhanced Integrated Device Electronics
EMI	Electromagnetic Interference
FDD	Floppy Disk Drive
FIFO	First In, First Out
FPGA	Field Programmable Gate Array
HCL	Hardware Compatibility List
HDD	Hard Disk Drive
IDE	Integrated Device Electronics
ISA	Industry Standard Architecture
LED	Light Emitting Diode
LFU	Loadable Firmware Update
LVD	Low Voltage Differential
LW	Late Write
MOP	Maintenance Operation Protocol
OEM	Original Equipment Manufacturer
NVRAM	Non-Volatile Random Access Memory
PAL	Privileged Architecture Library
PCI	Peripheral Component Interconnect
PIO	Programmed Input/Output
PLL	Phase Locked Loop
PCB	Printed Circuit Board
SCSI	Small Computer System Interface
SDRAM	Synchronous Dynamic Random Access Memory
SRM	System Reference Manual

Acronym	Meaning
SROM	Serial Read-only Memory
SRAM	Static Random Access 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, including its components and features.

The UP2000+ product consists of a UP2000+ Motherboard and one or two Alpha Slot B Modules. The Alpha Slot B Module consists of the 21264 microprocessor, Bcache (level 2 cache) and voltage converters to convert from 12 or 5 Volts to between 2.2 and 1.5 Volts.

Note: *The position in which you use an Alpha Slot B Module on the UP2000+ Motherboard, primary connection or secondary connection, determines the following:*

- *J23 provides the connection for the primary, or first, Alpha Slot B Module and runs on 12V*
- *J22 provides the connector for the secondary, or second, Alpha Slot B Module and runs on 5V*

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 either of the following types of processors: *Note: You must use only one type of processor on a motherboard. Do not mix types of processors on a motherboard.* • Alpha 21264/21264A processors, at speeds of 667 or 750 MHz • Alpha 21264B Double Data Rate (DDR) processor, at speeds of 833 MHz	API NetWorks, Inc.

Table 1-1 UP2000+ Product Features (Continued)

Feature	Description	Manufacturer
Chipset:	21272 (Tsunami)—One Cchip, four Dchips, and two Pchips provide the following: • Maximum 83 MHz system bus with Double Data Rate (DDR) transfers, maximum bandwidth of 2.67 GBytes/second • One 256-bit memory bus • Two 64-bit, 33 MHz PCI buses, with two 64-bit slots and one 32-bit slot on each bus	Compaq Computer Corporation
Cache:	External L2 cache with 128-bit data path for an Alpha Slot B Module supports: • 2 MB or 4 MB cache per 667 MHz processor, Late Write (LW) Synchronous SRAMS (SSRAMs) • 8 MB cache per 750 MHz processor, LW SSRAMs • 4 MB or 8 MB cache per 833 MHz DDR processor, DDR SRAMs	
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:	AT- or ATX-compatible power connector, supplying +3.3 Vdc, ± 5 Vdc, and ± 12 Vdc	
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 • AIC-7891 Small Computer System Interface (SCSI) Controller (Ultra2 SCSI), connections for up to 15 SCSI devices • 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	Cypress Adaptec SMC
I/O Slots:	• Four 64-bit, 33 MHz PCI Slots, *PCI Local Bus Specification Revision 2.1* compliant	
	• Two 32-bit, 33 MHz PCI Slots, *PCI Local Bus Specification Revision 2.1* compliant	
	• One ISA Slot (Shared)	

Table 1-1 UP2000+ Product Features (Continued)

Feature	Description	Manufacturer
Firmware:	• Embedded Alpha System Reference Manual (SRM) Console	
System Management (via PCI-ISA Bridge I^2C Controller or System Management Expansion Connector J40):	• Monitoring of processor and motherboard voltages and fan speeds • Monitoring of processor thermal state • Detection of processor and motherboard presence, versions, and asset record • Detection of system and power supply status and power supply inhibiting • Indication of system error for both hardware and software detected problems	

1.2 System Components

The UP2000+ is implemented in industry-standard parts and uses one or two 21264 central processing units (CPUs). The functional components of the UP2000+ are shown in block diagram form in Figure 1-1. A detailed description of system components is provided in Chapter 1, "Functional Description", of the *UP2000+ Technical Reference Manual* (P / N 51-0039).

Note: *Refer to the list of Acronyms on page x of the Preface for a definition of terminology used in the block diagram.*

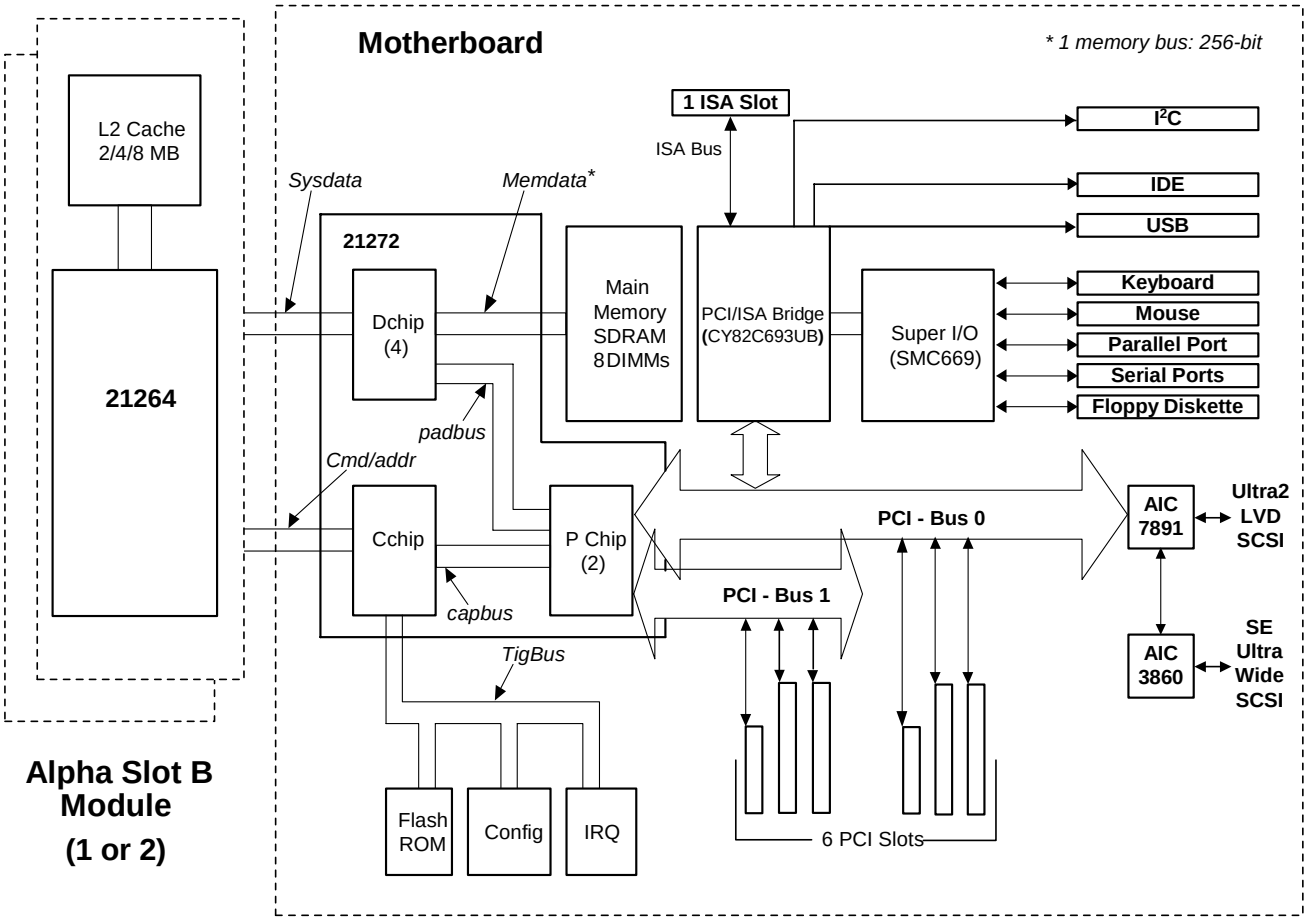

Figure 1-1 UP2000+ Functional Block Diagram

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 for:

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

All UP2000+ connectors and configuration jumpers are shown in Figure 2-1, which depicts the UP2000+ Motherboard and its components. Table 2-1 identifies the components shown in Figure 2-1. Appendix A contains a complete description of the connectors and pinouts used in the UP2000+.

2.1.1 Configuration Requirements

The UP2000+ automatically configures the L2 cache size, and the clock frequencies for the 21272 chipset and the 21264 CPU.

A two-pin jumper block, J31, selects the correct output voltage for the Alpha Slot B Module CPU. For a 21264/21264A processor, no jumper is required on J31. For a 21264B processor, a jumper is required on J31. Information is provided with each Alpha Slot B Module to identify which type of processor is present in the module.

Jumper block J29 makes use of manually inserted two-pin shunts to configure the UP2000+ for upgrade or error recovery functions. See section 5.2, “Error Recovery,” on page 5-4 for details on these procedures.

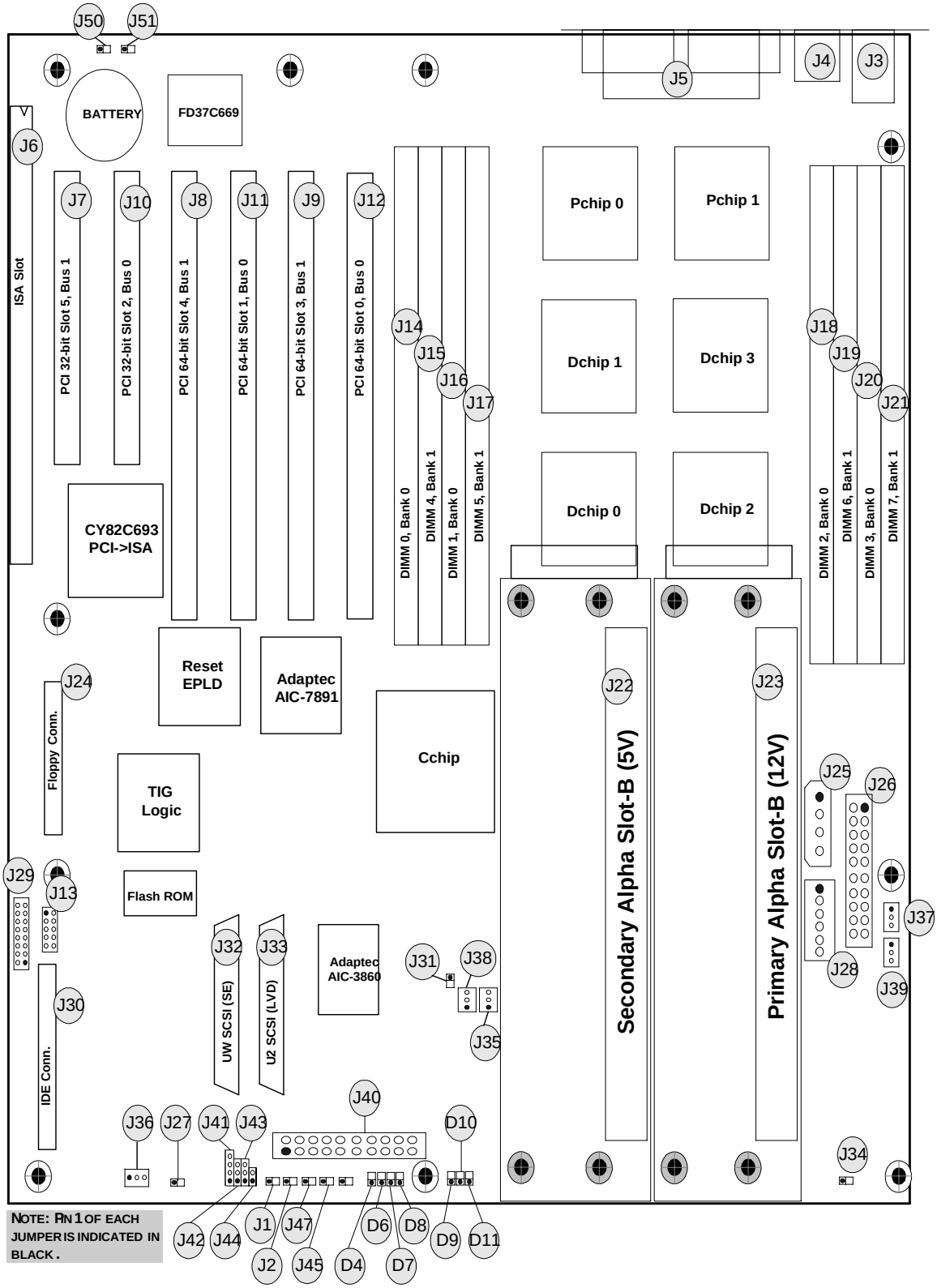

Figure 2-1 UP2000+ Motherboard Layout

Table 2-1 UP2000+ Motherboard Jumper and Connector Component List

Comp. No.	Specification	Comp. No.	Specification
J1	Power Good Light Emitting Diode (LED) Connector	J2	Hard Disk Drive (HDD) Activity LED 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	64-bit PCI Slot 4, Bus 1 Connector
J9	64-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, Bank 0
J15	168-pin DIMM 4 Socket, Bank 1	J16	168-pin DIMM 1 Socket, Bank 0
J17	168-pin DIMM 5 Socket, Bank 1	J18	168-pin DIMM 2 Socket, Bank 0
J19	168-pin DIMM 6 Socket, Bank 1	J20	168-pin DIMM 3 Socket, Bank 0
J21	168-pin DIMM 7 Socket, Bank 1	J22	Alpha Slot B Connector for Secondary Module (5V)
J23	Alpha Slot B Connector for Primary Module (12V)	J24	Floppy Disk Drive (FDD) Connector
J25	Alpha Slot B Module Power Connector (4 pin)	J26	ATX Power Connector (20 pin)
J27	System Fail LED Connector	J28	AUX ATX Power Connector (6 pin)
J29	Firmware Configuration Jumper	J30	IDE Bus Connector
J31	Processor Configuration Jumper	J32	Ultra-wide SCSI Single-ended (SE) Connector
J33	Ultra-wide SCSI Low Voltage Differential (LVD) Connector	J34	Buzzer Connector
J35	Secondary Fan Connector[1]	J36	AC Fail/Rack# Connector[2]
J37	Primary Fan Connector[1]	J38	Secondary Fan Connector[1]
J39	Primary Fan Connector[1]	J40	System Management (SM) Expansion Connector
J41	Speaker Connector	J42	Secondary Alpha Slot B Module Debug Port (SROM) Connector

Notes:
1. ***When you use the UP2000+ in the PowerRAC Chassis 320, connect J37 and J39 to the Alpha Slot B Module fans, and connect J35 and J38 to the PCI slot fans. When you use the UP2000+ in a pedestal enclosure, connect J37 and J39 to the primary Alpha Slot B Module fans, and connect J35 and J38 to the secondary Alpha Slot B Module fans.***
2. ***When you use the UP2000+ in the PowerRAC Chassis 320, connect J36 to the enclosure power supply with the supplied cable. When you use the UP2000+ in a pedestal enclosure, do not connect J36 to anything.***
3. ***If you use the UP2000+ in a pedestal enclosure., you may have optional system fans in the enclosure which you can connect using J50 and J51. Both connectors are limited to 1A@12V usage.***

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	System Power Button Connector
J45	Halt Button Connector	J46	Not Used
J47	Reset Button Connector	J50	Optional System Fan Connector[3]
J51	Optional System Fan Connector[3]		

Notes: ***1. When you use the UP2000+ in the PowerRAC Chassis 320, connect J37 and J39 to the Alpha Slot B Module fans, and connect J35 and J38 to the PCI slot fans. When you use the UP2000+ in a pedestal enclosure, connect J37 and J39 to the primary Alpha Slot B Module fans, and connect J35 and J38 to the secondary Alpha Slot B Module fans.***

2. When you use the UP2000+ in the PowerRAC Chassis 320, connect J36 to the enclosure power supply with the supplied cable. When you use the UP2000+ in a pedestal enclosure, do not connect J36 to anything.

3. If you use the UP2000+ in a pedestal enclosure., you may have optional system fans in the enclosure which you can connect using J50 and J51. Both connectors are limited to 1A@12V usage.

2.1.2 SCSI Connections

The UP2000+ supports two types of SCSI devices: Ultra-wide and Ultra2. Low Voltage Devices (LVD) are Ultra2 SCSI, while Single-Ended (SE) devices are Ultra-wide SCSI. The Ultra2 LVD SCSI controller is the Adaptec AIC-7891. An Adaptec AIC-3860 driver combines with the AIC-7891 controller to provide Ultra-wide SE SCSI.

Note: *SE SCSI use is not supported outside of the chassis in which the UP2000+ Motherboard is installed.*

2.1.3 Status Indicators

The LED status indicators (D4 and D6–D11) are described in Table 5-1, “LED Status Indicators,” on page 5-2.

2.2 Memory Configuration

The memory subsystem consists 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. See Figure 2-2 for slot locations.

Use the following rules for installing memory in the UP2000+:

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

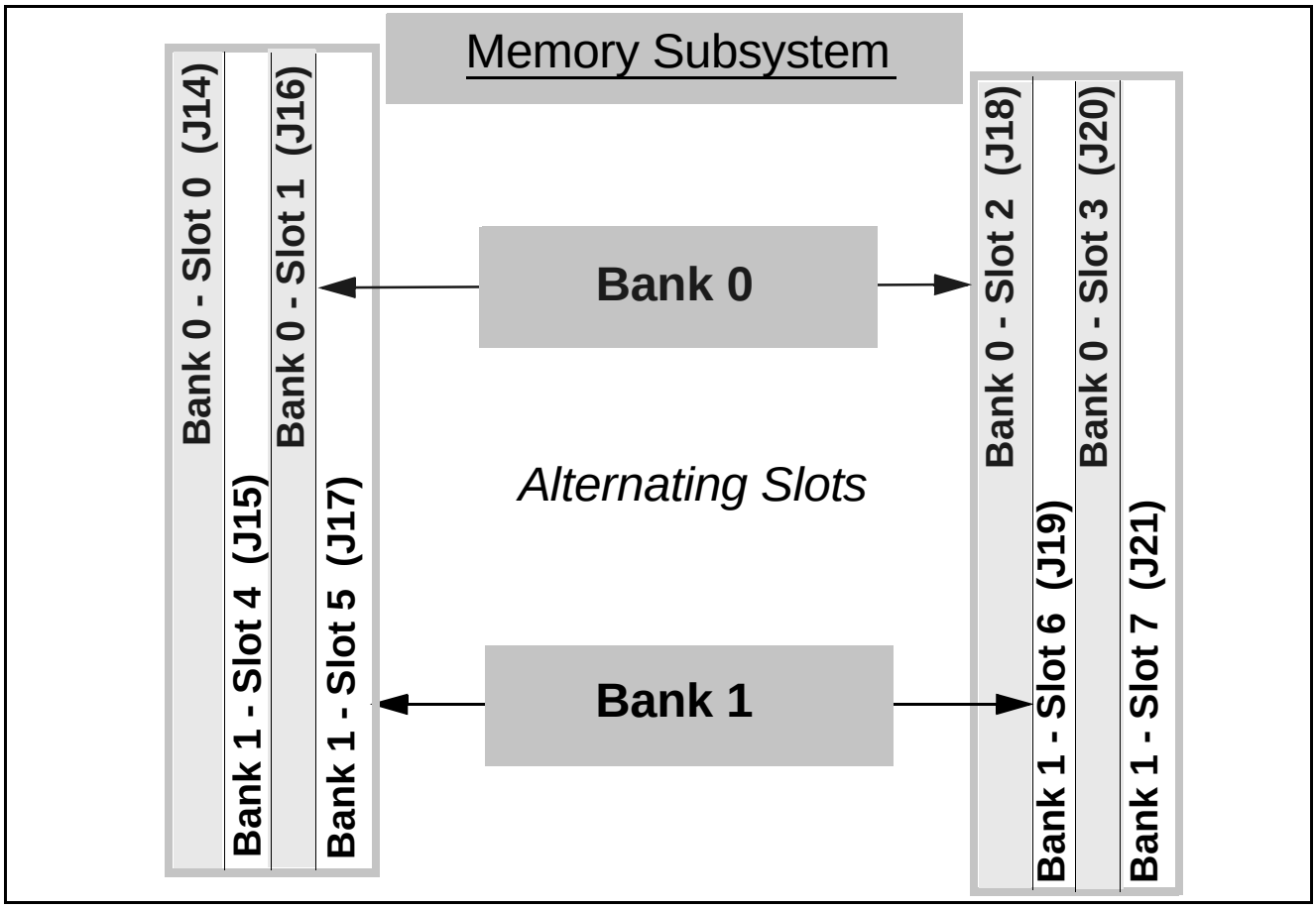

Figure 2-2 Memory Subsystem

2.3 Configuration Jumpers

The UP2000+ Motherboard has one two-pin jumper block, J31, used for CPU configuration. A second, programmable jumper block, J29, is used for firmware configuration. Both jumper blocks are located on the UP2000+ Motherboard as shown in Figure 2-3.

Note: *J29 is configured for normal operation when it leaves the factory. Do not change the factory settings to use this product.*

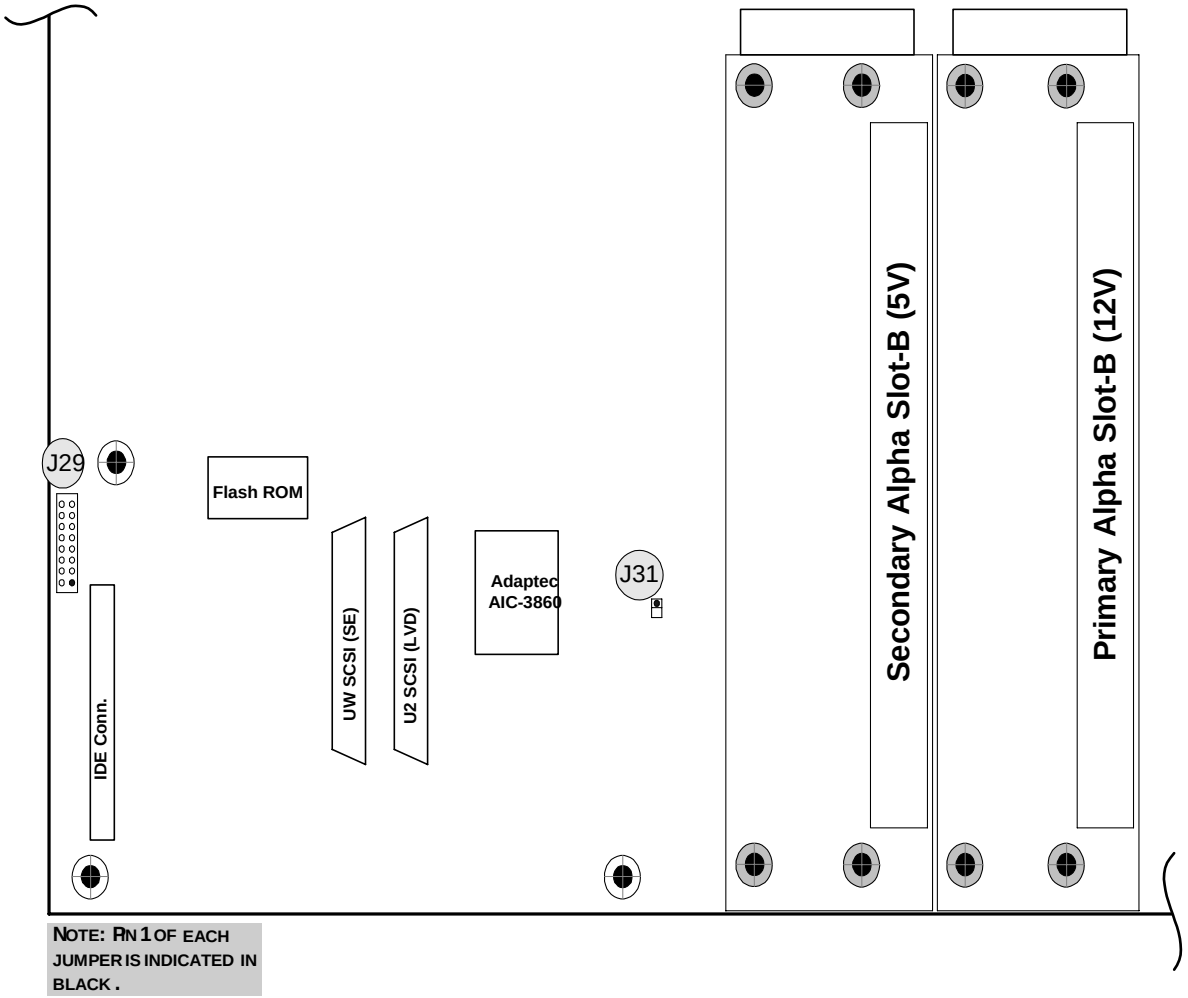

Figure 2-3 Configuration Jumper Blocks

2.3.1 Processor Voltage Selection

The Alpha Slot B Modules used on the UP2000+ contains either a 21264/21264A or a 21264B processor. J31 allows selection of the proper Voltage Regulation Module (VRM) output voltage for the processor chip

on the Alpha Slot B Module, depending on the generation of chip used. Settings for J31 are as follows:

- Do not install a shunt on J31 for a 21264/21264A processor
- Install a shunt on J31 for a 21264B processor

Information is provided with each Alpha Slot B Module to identify which generation of processor is present in the module.

Note: *J31 configures the output voltage for both the primary and secondary Alpha Slot B Modules. A dual-processor system must use the same type, size and speed of Alpha Slot B Modules in both slots.*

2.3.2 Serial Remote Reset Disable

Configuration of the serial remote reset function is managed through pins 13 and 14 of J29. By default, a jumper is installed on pins 13 and 14 of J29, and the serial remote reset function is enabled. To disable serial remote reset, remove the jumper from pins 13 and 14.

2.3.3 Diagnostics and Flash Recovery

Configuration of diagnostics and flash recovery is managed through settings of pins 1–8, 15 and 16 of J29. Table 2-2 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 any pins of J29.

Table 2-2 Diagnostics and Flash Recovery Configuration Settings (J29)

Function	Install Jumper on Pins:				
	1/2	3/4	5/6	7/8	15/16
Normal Boot Sequence (using firmware)—Default	0	0	0	0	0
Restore Factory Defaults (Bypass password check)	0	0	0	0	1

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 SROM code function, type an uppercase U to set the interactive baud rate higher.***
3. ***Refer to the* UP2000+ Technical Reference Manual*, P/N 51-0039, for information on SROM code and Alpha Diagnostics.***

Table 2-2 Diagnostics and Flash Recovery Configuration Settings (J29) (Continued)

Function	Install Jumper on Pins:				
	1/2	3/4	5/6	7/8	15/16
Normal Recovery (using firmware)	1	0	0	0	0
Alpha Diagnostics Recovery (using firmware)	1	0	0	0	1
Interactive Alpha Diagnostics (using operator console)	1	0	0	1	0
Interactive Alpha Diagnostics (using Debut Port J42/J43)	1	0	0	1	1
Interactive SROM code (using Debug Port J42/J43)	0	0	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 SROM code function, type an uppercase U to set the interactive baud rate higher.

3. Refer to the* UP2000+ Technical Reference Manual*, P/N 51-0039, for information on SROM code and Alpha Diagnostics.

2.3.4 Flash Write Protection

The UP2000+ provides write protection for the firmware flash ROM. By default, a jumper is not installed on J29, pins 9 and 10 of the UP2000+ Motherboard, and write protection to the flash ROM is not enabled.

Install a jumper on J29, pins 9 and 10 to enable write protection.

2.3.5 Cache Disable

By default, a jumper is not installed on pins 11 and 12 of J29, and the UP2000+ automatically reads L2 cache size from the Alpha Slot B Modules.

Install a shunt on J29, pins 11 and 12, to instruct system firmware to override Alpha Slot B Module inputs and set the L2 cache size to 0.

Chapter 3 Electrical and Physical Data

This chapter provides a description of the UP2000+ power requirements, environmental and enclosure specifications, and physical parameters.

3.1 Power Specifications

3.1.1 Power Connectors

The power connectors specified to support dual Alpha Slot B Modules are shown in Figure 3-1, and are described as follows:

- ATX Standard: One 10 x 2 (20-pin)
- AUX ATX: One 6 x 1 (6-pin)
- Alpha Slot B Module Power: One 4 x 1 (4-pin)

Either 5V or 12V is input to the Unitrode UCC3882-1DW VRM, which outputs the core voltage to the CPU. Because over 9A is required by the Alpha Slot B Modules, one Alpha Slot B Module Power 4 x 1 (4-pin) connector is added. To support the 3.3V draw, an AUX ATX 6 x 1 (6-pin) connector is used.

Note: *Use the AUX ATX 6-pin connector when an additional 6-pin connection is included with the ATX Standard power connector. The AUX ATX connector mates with a Molex 90331-0010 or equivalent part.*

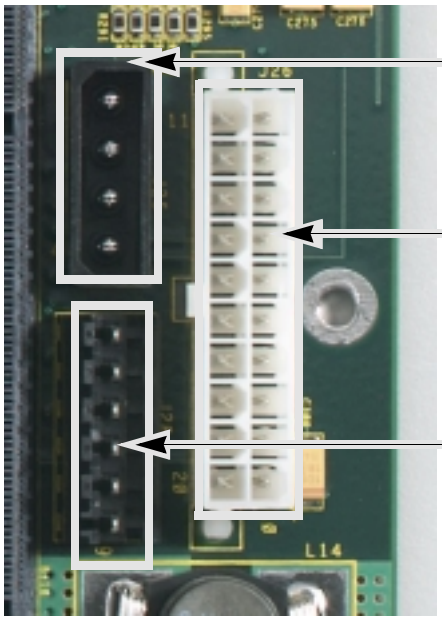

Figure 3-1 Power Connectors

3.1.2 Power Supply

Pedestal Enclosure

The UP2000+ requires the use of a 600 Watt ATX power supply, as described in Application Note 51-0038-0A, *MACASE Chassis RFI Upgrade Kit*.

API NetWorks, Inc, recommends the power supply described in Table 3-1, or any comparable power supply which can provide the same level of support.

Table 3-1 Recommended Power Supply

Feature	Specification
Vendor and Model:	PC Power and Cooling, Inc. Model Turbo-Cool 600, T60R-X
Output:	70A @ +5V 25A @ +12V 1A @ -5V 1A @ -12V 50A @ +3.3V 0.85A @ 5Vsb

Table 3-1 Recommended Power Supply (Continued)

Feature	Specification
Qualifications:	• Maximum allowable 3.3V + 5V total draw = 350W • Maximum total continuous power = 500W • Maximum total peak power = 700W

PowerRAC Chassis 320 Enclosure

When installed in a PowerRAC Chassis 320, the UP2000+ uses the chassis's semi-custom 600 Watt power supply.

3.1.3 Estimated Maximum Power Consumption

The UP2000+ Motherboard has a maximum total power consumption of 97.05W. The primary Alpha Slot B Module has a maximum total power consumption of 165.8W, while the secondary Alpha Slot B Module has a maximum total power consumption of 142W.

Table 3-2 lists the current requirement for each direct current supply voltage (Vdc) for the UP2000+ Motherboard. Table 3-3 and Table 3-4 list the current requirement for each direct current supply voltage (Vdc) for the primary and secondary Alpha Slot B Modules, respectively. All requirements assume maximum usage applied.

Note: *These tables do not include requirements for SDRAM, peripheral slots or disk drives. Be sure to allow for adequate additional current when selecting a power supply for the UP2000+.*

Table 3-2 Estimated Power Consumption–UP2000+ Motherboard

V_{DD} Source	Current	Power	Remarks
3.3V	17.2A	56.8W	
5V	6.8A	34.0W	
-5V	0.05A	0.25W	
12V	0.4A	4.8W	Fans
-12V	0.1A	1.2W	
Total Power Consumption:		97.05W	

Table 3-3 Estimated Power Consumption–Primary Alpha Slot B Module, 8 MB L2 Cache

V_{DD} Source	Current	Power	Remarks
3.3V	6.3A	20.8W	
5V	0.2A	1.0W	
12V	12A	144W	VRM
Total Power Consumption:		165.8W	

Table 3-4 Estimated Power Consumption–Secondary Alpha Slot B Module, 8 MB L2 Cache

V_{DD} Source	Current	Power	Remarks
3.3V	6.3A	20.8W	
5V	24A	120W	VRM
12V	0.1A	1.2W	
Total Power Consumption:		142.0W	

3.2 Environmental Specifications

The UP2000+ Motherboard and Alpha Slot B Module are specified to run within the environment listed in Table 3-5.

Table 3-5 Environmental Requirements

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

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 EN60950:1992 "Safety of Information Technology Equipment Including Electrical Business Equipment Incorporating Amendment Nos 1, 2, 3, 4." European Norm (EN) standards are published in the Official Journal of the European Community.

3.2.2 EMI

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

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: *These certifications were met with the UP2000+ installed in a PowerRAC Chassis 320 enclosure.*

The UP2000+ also meets the EMI immunity certification EN 50082-1:1992 "EMC Residential, Commercial and Light Industrial Generic Immunity Standard."

3.2.3 Thermal

Figure 3-2 shows the location of thermally-sensitive components on the UP2000+ Motherboard. A list of maximum allowable case temperatures for these components is provided in Table 3-6.

Case temperatures are a vital factor in determining airflow on a motherboard. Variables which may affect a component's case temperature include the following:

- Operating temperature
- Operating frequency
- Current load

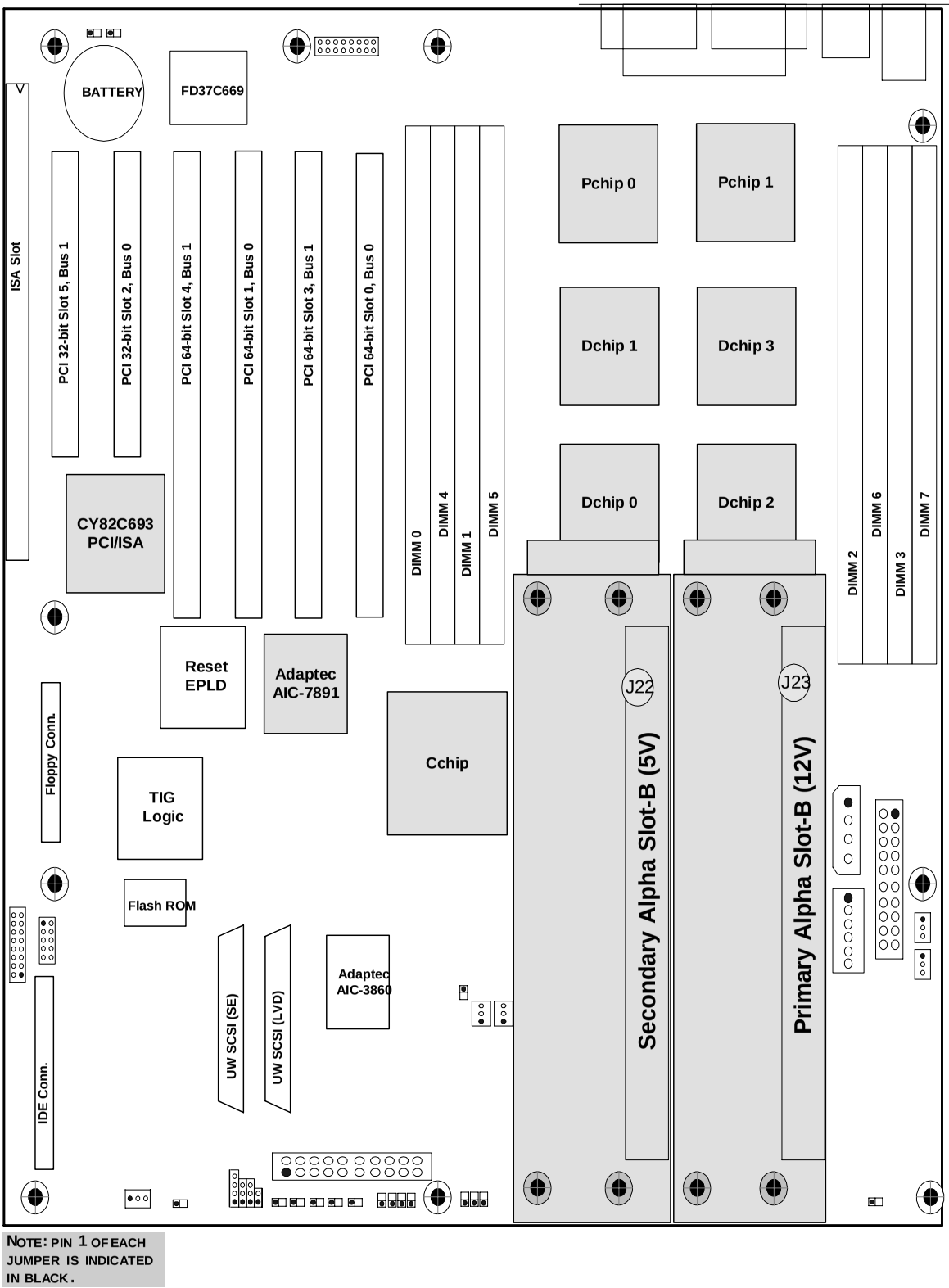

Figure 3-2 Thermally-sensitive Components

Table 3-6 Maximum Component Case Temperatures

Component	Maximum Temperature
Primary Alpha Slot B Module*	<70°C (158°F)
Secondary Alpha Slot B Module*	<64°C (147°F)
21272 D-chips	<69°C (156°F)
21272 P-chips	<66°C (151°F)
21272 C-chip	<54°C (129°F)
Adaptec AIC-7891 SCSI Controller	<69°C (156°F)
approved Southbridge peripheral bus controller	<55°C (131°F)
Note: * Data from results of test using 667 MHz, 4 MB Alpha Slot B Modules. All other components' data from results of test using 750 MHz, 8 MB Alpha Slot B Modules.	

Thermal tests conducted by API NetWorks, Inc, determined that these components remained within maximum specified temperatures. The test system configuration comprised the following:

- Dual 750 MHz, 8 MB Alpha Slot B Modules
- Fully-populated I/O and memory
- Minimum firmware to boot operating system (OS)
- 35°C ambient chamber

A thermocouple was placed on the base of each Alpha Slot B Module's heat sink, between the heat-sink mounting studs. All other components were instrumented at the top center of the case.

3.3 Enclosure Requirements

Enclosures used with the UP2000+ must have the following capabilities:

- Accept an Extended ATX form factor motherboard
- Accommodate eight goalpost assembly mounting holes

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

WARNING: Alpha Slot B Module goalpost assemblies must be securely fastened to the chassis backplate. Refer to the** UP2000+ Quick Start Installation Guide, P/N 51-0041**, or the** PowerRAC Chassis 320 Quick Start Installation Guide, P/N 51-0040**, for complete mechanical installation details. Failure to correctly attach the goalpost assemblies to the chassis backplate may damage the motherboard and may affect the product warranty.

3.3.1 Pedestal Installations

Note: *See the UP2000+ Quick Start Installation Guide, P/N 51-0041, for step-by-step instructions on how to install a UP2000+ Motherboard in a pedestal chassis. Do not attempt to install the UP2000+ Motherboard in a chassis without using the UP2000+ Quick Start Installation Guide.*

Chassis Mounting Holes

Before you begin installation of the UP2000+ Motherboard, note the following:

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.api-networks.com/products/downloads/customer_support/UP2000/UP2000_drill_template.pdf`

Required Equipment

When using a pedestal enclosure, the requirements for attaching the motherboard and goalposts fixture to the chassis are as follows:

- Fastening Hardware—
  - Eight Alpha Slot B hex male/female spacer screws and spacer washers
  - Eight M4X6 mm Phillips pan-head 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 inch)
- Proper chassis mounting holes drilled for goalpost captive nuts

Assembly Procedure

Use the following mounting technique to install a UP2000+ Motherboard in a pedestal enclosure:

1. Install the goalpost with fan cables labeled J37 and J39 over the primary Alpha Slot B Connector (J23).

   Install the goalpost with fan cables labeled J35 and J38 over the secondary Alpha Slot B Connector (J22).
2. Align the goalposts fixture holes with the corresponding motherboard clearance holes.
3. Affix goalposts to motherboard from the backside using hex male/female spacer screws and spacer washers. 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 goalpost brace to top left corners of both goalpost's fan mounting plates with M3x8mm pan-head Phillips SEMS lock and flat. Torque to 4 inch/lbs.

Note: *When you use the UP2000+ in a pedestal enclosure, do not connect J36. The status of this connector sets the motherboard signal RACK#. In a pedestal enclosure, leaving J36 unconnected set the RACK# signal to 1, which tells the system that it is not in a rackmount configuration.*

3.3.2 PowerRAC Chassis 320 Installations

To create a high-density system, you can install the UP2000+ Motherboard into a rackmount chassis. Multiple chassis can be installed in a single 19-inch rack.

Note: *The PowerRAC Chassis 320 Quick Start Installation Guide, P/N 51-0041, contains step-by-step instructions on how to install a UP2000+ Motherboard in a low profile (3U) chassis. Do not attempt to install the UP2000+ Motherboard in the PowerRAC Chassis 320 without using the PowerRAC Chassis 320 Quick Start Installation Guide.*

Following is an overview of the assembly process:

1. Remove the chassis top panel and subassemblies, including the Goalpost and Fan Assemblies, Hard Disk Drive Carrier, and the Floppy Disk/CD-ROM Combination Drive Carrier.
2. Assemble the drives into the chassis.
3. Install the UP2000+ Motherboard.
4. Install the Power and Internal I/O cables and the Fan Assembly.
5. Install modules and replace top panel.
6. Assemble rack mounting slide kits and install chassis in rack.

Note: *When you use the UP2000+ in the PowerRAC Chassis 320, connect J36 to the enclosure power supply with the supplied cable. Connecting J36 sets the motherboard signal RACK# to 0, which tells the system that it is in a rackmount configuration.*

3.4 Physical Parameters

3.4.1 UP2000+ Motherboard Parameters

The UP2000+ Motherboard is a printed circuit board (PCB) with the dimensions specified in Table 3-7.

Table 3-7 UP2000+ Motherboard Physical Parameters

Dimension	Value
Length	330.2 mm (13 in)
Width	304.8 mm (12 in)
Height	1.6 mm (.06 in)

3.4.2 Alpha Slot B Module Parameters

The Alpha Slot B Module is a PCB with the dimensions specified in Table 3-8.

Table 3-8 Alpha Slot B Module Physical Parameters

Dimension	Value
Length	168.8 mm (6.6 in)
Width	47.8 mm (1.9 in)
Height	114.3 mm (4.5 in)
Weight	1.2 kg (2.6 lb)

3.4.3 UP2000+ Motherboard Mounting Hole Specifications

The UP2000+ Motherboard mounting hole specification is depicted in

Figure 3-3. Note the goalpost assembly mounting area.

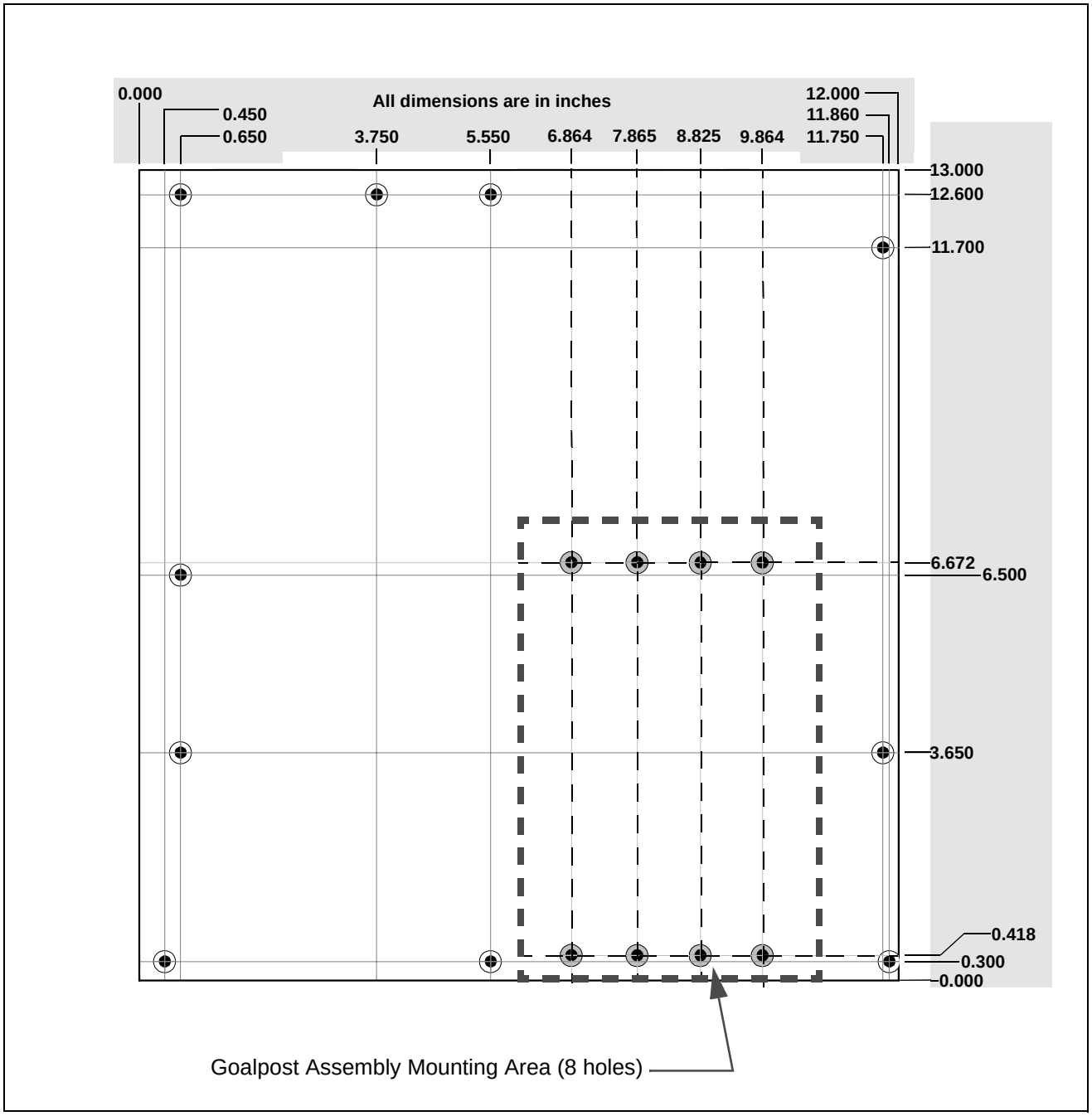

Figure 3-3 Mounting Hole Specifications

3.4.4 I/O Shield

The rear-panel connectors must be fitted with a suitable ATX I/O shield. The UP2000+ uses ATX Core Design #1 I/O shield, as defined in *Design Guide for the Intel ATX Motherboard I/O Implementation*, Version 1.1 by Intel Corporation. A copy of this design guide can be found on the following website:

```
http://www.teleport.com/~ffsupprt/
spec/atx/atxio_11.pdf
```

The UP2000+ rear-panel connectors are listed in Table 3-9.

Table 3-9 Rear Panel I/O Connectors

Component	Connector Description
J3	PS/2 Stacked Mouse/Keyboard (DIN)
J4	Dual Stacked USB
J5	Stacked Parallel/Serial I/O (25 Pin D-Sub)

Chapter 4 Software Support

4.1 Software Overview

UP2000+ systems support two major software components:

- SROM code
- Alpha System Resource Manual (SRM) Console
- Operating System (OS)

4.1.1 SROM Code

When the UP2000+ is turned on or reset, SROM code firmware automatically loads performs system initialization activities. Once SROM code firmware is loaded, it automatically load the next level of firmware and passes control to that code.

The UP2000+ supports SROM code version 1.9.3 or higher.

4.1.2 Alpha SRM Console

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

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

`http://www.api-networks.com/support/ srm-howto.asp`

The UP2000+ supports SRM Console version A5.5-82 or higher.

4.1.3 Operating System

The UP2000+ works with the Linux kernel 2.2.14 or higher.

Note: *Consult the API website 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. If the system is already running, you can invoke the SRM Console by:

1. Shutting down the operating system
2. Pressing the Halt button

Both of these actions return you to the SRM Console prompt, `P00>>>`.

For example, in a running system in which control has been passed to the Linux operating system, do one of the following steps to invoke SRM Console mode:

1. Shut down the operating system according to the procedure described in your operating system documentation. The SRM Console prompt, `P00>>>`, appears.
2. Press the Halt button. The SRM Console prompt, `P00>>>`, appears.

To return to the operating system:

At the `P00>>>` prompt, type **`boot`**

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 describes the most commonly used SRM Console commands. Table 4-2 defines the syntax for the SRM Console commands. Table 4-3 lists 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.

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

Command	Function
show pal	Displays the version of the privileged architecture library code (PALcode).
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 SRM Console Commands Syntax

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.
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 radic 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.

Table 4-2 SRM Console Commands Syntax (Continued)

Option	Attribute or Action
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 SRM Console Special Characters

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 overstrike.
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.
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.

Table 4-3 SRM Console Special Characters (Continued)

Character	Function
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

                    API UP2000+ 667 MHz
SRM Console:  VA5.5-82
PALcode:      OpenVMS PALcode V1.61-49, UNIX PALcode V1.54-50
Processors
CPU 0         Alpha 21264-3 500 MHz SROM Revision: V1.93
              Bcache size: 4 MB
CPU 1         Alpha 21264-3 500 MHz SROM Revision: V1.93
              Bcache size: 4 MB
```

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: X1.82 X1.93

P00>>>
```

show device

The **`show device`** command displays status for devices and controllers in the system, including 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
dkc0.0.0.9.0        DKC0
dkc100.1.0.9.0      DKC100
dkc200.2.0.9.0      DKC200
dkc300.3.0.9.0      DKC300
dkc500.5.0.9.0      DKC500
dva0.0.0.0.0        DVA0
ewa0.0.0.8.1        EWA0          00-00-F8-00-0E-3B
pkc0.7.0.9.0        PKC0          SCSI Bus ID 7 5.54
P00>>>
```

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	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	One-letter designator of storage adapter (a, b, c, and so on).
200	Device unit number. A unique MSCP unit number. SCSI unit numbers are forced to 100 X node ID.
2	Bus node ID number.
0	Channnel number. Used for multi-channel devices.
7	Logic slot number. Corresponds to PCI slot number.
1	Host 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      128 MB       000000000
1      128 MB       008000000
Total Bad Pages = 0
Total Good Memory = 512 MBytes
P00>>>
```

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, UNIX PALcode V1.54-50 Processors
P00>>>
```

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 0      good
Power Supply 1/Fan Tray good
System Fans         good
CPU Fans            good
Temperature         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 JUL 21 1999 16:39:43
P00>>>
```

4.2.4 Booting the Operating System

The **boot** command is used to boot the operating 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
```

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_dev*]**

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 4-15 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.
Note: The operator console is the monitor, keyboard, and mouse. This hardware is used to enter SRM Console commands into the system.	

Table 4-6 Boot Command Options (Continued)

Option	Description
-protocols *enet_protocol*	Either mop or bootp (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 4-15). 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.	

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.*

To run LFU, set the auto_action *variable to* halt*, then reset the system. Remember to reset* auto_action *to the original value after you run LFU.*

Example 4-9 `lfu` Command

```
P00>>> lfu

***** Loadable Firmware Update Utility *****

Select firmware load device (cda0, dva0, ewa0), or
Press <return> to bypass loading and proceed to LFU: cda0


Please enter the name of the options firmware files list, or
Press <return> to use the default filename [AS1400FW]: AS1400CP

Copying AS1200CP from DKA500.5.0.1.1 .
Copying [as1200]TCREADME from DKA500.5.0.1.1 .
Copying [as1200]TCSRMROM from DKA500.5.0.1.1
.......................
Copying [as1200]TCARCROM from DKA500.5.0.1.1 ............
```

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 4-22 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` command. See section 4.2.8 on page 4-22 for further information.

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 Creating a User-Defined Environment Variable

```
P00>>> edit nvram
editing 'nvram'
0 bytes read in
*10 set mopv3_boot 1
*exit
17 bytes written out to nvram
P00>>>
```

The nvram script is edited so an environment variable called `mop3_boot` is created and set to 1 on each power-up. By default, MOP boots send four MOP V4 requests before defaulting to MOP V3. This user-created environment variable forces the SRM Console to bypass MOP V4 requests, speeding up MOP booting on networks with MOP V3 software.

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
a	Load operating system software from the specified boot device (autoboot). Boot to multi-user mode.
i	Prompt for the name of a file to load and other options (boot interactively). Boot to single-user mode.
s	Stop in single-user mode. Boots to single-user mode and stops at the # (root) prompt.

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.

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`).

You will be asked to confirm the change, as shown in the following example.

Example 4-12 Using com*_baud

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

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 **console**, 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

The **ew*0_protocols** variable 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_type**, 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 **language**, 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_type**, 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

The **set password** command sets or clears the SRM Console password stored in Non-Volatile RAM (NVRAM).

The syntax is: **set password**

A 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 in length.

pci_parity

To disable or enable parity checking on the PCI bus, use the **`set pci_parity`** command.

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

To set the controller host bus node ID to a value between 0 and 7, enter the **`set pk*0_host_id`** command.

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 4-7). 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_id`**, 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

The **`pk*0_soft_term`** command 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 the `set console` description on page 4-16) 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 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.

Example 4-13 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]
```

4.3 Installing the Linux Operating System

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

As noted at the beginning of the chapter, this system supports many OS distributions and vendors. Consult our website for a complete current list at:

`http://www.api-networks.com/`

1. With your browser pointed at the API NetWorks, Inc. website, click on API Partner Program.
2. Scroll down in this page to the link, Linux Solutions Datasheet. Click on this link. This takes you to a datasheet on supported operating systems and applications.
3. Or, click on Software Vendors at the left center of the browser.

Examples of installing commercially available Linux distributions are given in this section. Typical requirements and procedures for Red Hat Linux Versions 6.2 or SuSE Version 6.3 follow.

4.3.1 Requirements

The following hardware components and settings are necessary for a Linux installation:

- Hard disk connected to the primary IDE port as a master, i.e., `/dev/hda`
- CD-ROM drive connected to the secondary IDE port as a master, i.e., `/dev/hdc`

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
- IDE CD-ROM names are "`/dev/hd`***x***", where ***x*** is the drive position
- SCSI hard disk names are "`/dev/sd`***x***", where ***x*** is the drive position
- IDE hard disk names are "`/dev/hd`***x***", where ***x*** is the drive position
- Floppy disk names are "`/dev/fd`***x***", where ***x*** is the device number

The following disks are required:

- The appropriate Linux operating system distribution CD-ROM disk, either the Red Hat Version 6.2 or SuSE Version 6.3 CD-ROM disk.

- If you are installing SuSE Version 6.3, you will need a ramdisk floppy. See the SuSE Version 6.3 documentation for instructions about creating this disk.

Note: *For all Linux distributions, be careful when configuring the X server. Do not test the X server configuration during the installation process.*

4.3.2 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`** (IDE disk) or **`DKA0`** (SCSI disk)
`boot_osflags`	Information passed to Linux kernel via boot Example: **`"root=/dev/hda2"`**
`boot_file`	Sets file to use as the kernel on the default boot device Example for Red Hat 6.2: **`kernels/generic.gz`**

From the SRM Console prompt, `P00>>>`, perform the procedures described in the following paragraphs.

- To set the default boot device:
  - For SuSE Version 6.3, to boot from a floppy drive:

    Type **`set bootdef_dev dva0`**
  - For Red Hat 6.2, to boot from an IDE CD-ROM drive:

    Type **`set bootdef_dev dqb0`**

  Note:*If your drive is a different device type, set this variable appropriately.*
- To set the default boot file to the kernel, choose the command that corresponds to your version of Linux:
  - For SuSE Version 6.3:

    Type **`set boot_file vmlinux.gz`**
  - For Red Hat 6.2:

    Type **`set boot_file kernels/generic.gz`**

- To set the default flags to point to the (currently non-existent) root partition:
  - On a SCSI hard drive:

    Type **`set boot_osflags "root=/dev/sda2"`**
  - On an IDE drive:

    Type **`set boot_osflags "root=/dev/hda2"`**

  Note: *Setting the flags to point to sda1 (Linux terminology for SCSI Disk A Partition 2) 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*`**

4.3.3 Starting the Linux OS Installation

To start the Linux OS installation, perform the following:

1. Insert the Red Hat Version 6.2 or SuSE Version 6.3 Alpha CDROM into the drive.
2. From the SRM Console prompt (`P00>>>`):
   - For Red Hat Version 6.2:

     Type **`boot -flags "root=/dev/hdc"`**

     Note: *If you are installing with the Anaconda graphical installation tool and you get a graphic card problem, add the "text" flag to install using the text-mode installation tool. Type the following to add the correct flag:*

     `P00>>>` **`boot -flags "root=/dev/hdc text"`**
   - For SuSE Version 6.3:

     Type **`boot -flags "root=/dev/fd0 load_ramdisk=1 ramdisk_size=20480"`**
3. Follow the instructions for your distribution.
4. When installation is complete, the system automatically begins to reboot. Wait until the system reaches the SRM Console prompt, then press **Ctrl/C** to stop the reboot process.

Background Information

To the SRM Console, the IDE CD-ROM drive is called `DQB0`. To the Linux kernel, it is called `/dev/hdc`. This means that the CD-ROM is the master device on the secondary IDE channel.

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/hdc"` to begin, you bypass the environmental setting made in section 4.3.4.

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 an SRM-bootable diskette or hard disk are available on-line at:

`http://www.alphalinux.org/faq/srm.html`

Post Installation Check

At this point, Linux is installed on the hard disk. If Partition 2 is the root directory, then the environmental variable **`boot_osflags`** is correct. If your root directory has another name, take this opportunity to set the variable to the corresponding name.

4.3.4 After Installing Linux

After installing the operating system, you must do the following:

- Copy the kernel to hard disk
- Reset environment variables

The following sections describe these procedures.

Copy Kernel to Hard Disk, Red Hat Versions 5.2, 6.0, and 6.1 or SuSE Versions 6.1 and 6.3

Use the following steps to copy the kernel:

1. Type the following at the SRM Console prompt `P00>>>`:
   `P00>>>` **`boot dva0 -fi vmlinux.gz -fl "root=/dev/hda2 single"`**
   The shell prompt appears.
   Note:*On a SCSI hard drive, use the value "***`root=/dev/sda2 single`***"*
2. Type the following commands at the bash# prompt:
   - For Red Hat—

     ```
     bash# mount -t ext2  /dev/fd0  /mnt/floppy
     bash# cp /mnt/floppy/*  /boot/
     ```

   - For SuSE—

     ```
     bash# mount -t ext2  /dev/fd0  /floppy
     bash# cp /floppy/*  /boot/
     ```

3. Type the following commands at the bash# prompt:

   ```
   bash# gzip -dc  /boot/System.gz >
        /boot/System.map-2.2.15
   bash# rm /boot/System.map
   bash# ln -s /boot/System.map-2.2.15 /boot/System.map
   bash# ln -s Tsunami  /etc/alpha_systype
   ```

4. To configure the monitor and video driver, perform the following at the bash# prompt:
   - For Red Hat—

     Type **`Xconfigurator`**

     Follow the `Xconfigurator` prompts to set up the monitor and video drivers.
   - For SuSE—

     Type **`sax`**

     Follow the `sax` prompts to set up the monitor and video drivers.
5. When the monitor and video drivers are configured, shutdown the system. Type the following command at the bash# prompt:

   ```
   bash# shutdown -h now
   ```

Reset Environmental Variables for Normal Boot

Reset the SRM Console environmental variables for normal operations after installing the OS.

- If you are using a SCSI disk, type the following at the SRM Console prompt, `P00>>>`:

  ```
  1. P00>>> set bootdef_dev dka0
  2. P00>>> set boot_file vmlinux.gz
  3. P00>>> set boot_osflags "root=/dev/sda2"
  4. P00>>> boot
  ```
- If you are using an IDE disk, type the following at the SRM Console prompt, `P00>>>`:

  ```
  1. P00>>> set bootdef_dev dqa0
  2. P00>>> set boot_file vmlinux.gz
  3. P00>>> set boot_osflags "root=/dev/hda2"
  4. P00>>> boot
  ```

Note: *When your OS installation is complete and working satisfactorily, set the* `auto_action` *variable to* `boot` *to boot into the OS automatically at power on.*

Reference Information

Refer to the README.TXT and INSTALL.TXT files on the *GPL Source Diskette* for other post-configuration information.

Chapter 5 Troubleshooting

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

Topics covered in this chapter include the following:

- Video review checklist
- Status LEDs
- Power On Self Test (POST) codes
- Beep codes
- Error recovery

5.1 Hardware Startup

5.1.1 No Video Present

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.).

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

4. Turn the system power OFF.
5. Reseat the video card and ensure that it is connected to the monitor.
6. Replace the video card.
7. Reseat the DIMMs.
8. Replace the DIMMs.

5.1.2 LED Status Indicators

Seven 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, to the left of

the Secondary Alpha Slot B Module. Their orientation is shown in Figure 5-1.

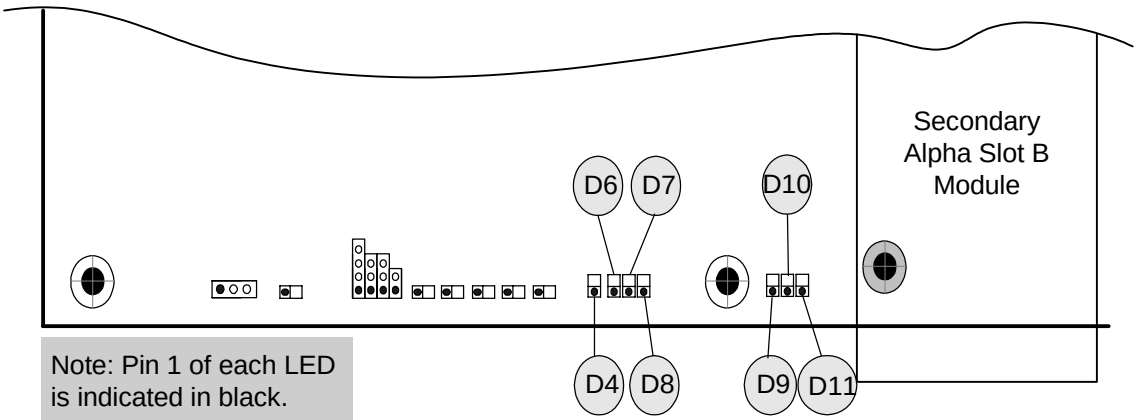

Figure 5-1 LED Locations

Table 5-1 provides information to help you interpret the LED status information

Table 5-1 LED Status Indicators

LED	Function		Comment
D4	UP2000+	PSU P_OK	ON when output power is good
D6	Secondary Alpha Slot B Module Debug Port	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
D9	Primary Alpha Slot B Module Debug Port	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 Codes

Using an ISA-based POST card module, you can monitor the sequential steps as the system initializes and loads 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 Descriptions

Source	POST Code (hex)	Message
SROM code	20	Firmware initialization complete
	01	CPU speed detected; initialize PCI-to-ISA Bridge
	02	PCI-to-ISA Bridge 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 Code Descriptions

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

5.2 Error Recovery

The UP2000+ Motherboard has a firmware configuration jumper block (J29) located on its left side just above J30, the IDE device port. It is an eight-position, 2-pin jumper type of block, as show in Figure 5-2. This jumper bock is used for troubleshooting and error recovery.

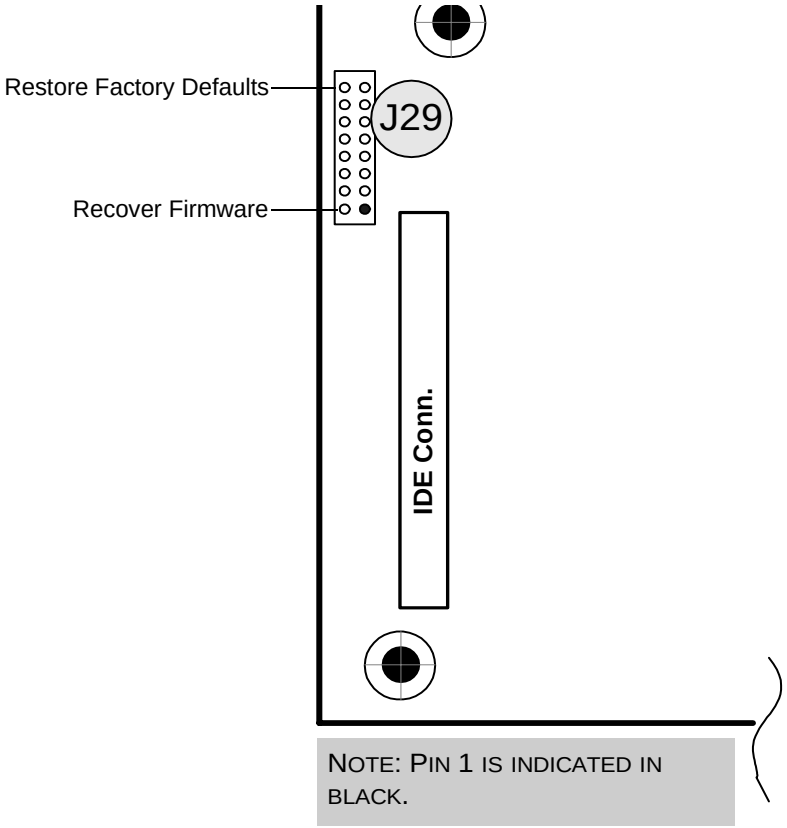

Figure 5-2 J29 Firmware Configuration Jumper Block

5.2.1 Error Conditions

In addition to the factory default configuration setting, there are two other configuration settings which you can use to recover from firmware errors on the UP2000+.

For more information, see the FAQs on the API NetWorks, Inc. website:

`http://www.api-networks.com`

Restore Factory Defaults

Some error conditions of this type include:

- Choosing incorrect selections when configuring the SRM Console. These selections may prevent the system from booting.
- Forgetting your system password.

For recovery from these types of errors, install a jumper on pins 15 and 16 of J29 to invoke the fail-safe boot recovery procedure.

Reload Firmware

An error condition of this type may occur during the upgrading of the SRM Console (see section 4.2.5 on page 4-11) by an improper system action. An example would be: accidentally powering off the system during this procedure.

For this type of recovery, install a jumper on pins 1 and 2 of J29 to override these errors.

5.2.2 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 shunt according to the error to be cleared (see Table 5-4).
3. Start the system.
4. Enter the proper parameters in Alpha SRM Console, then load the operating system.
5. Power OFF the system.
6. Restore the jumper shunts to their default positions.
7. Start the system.

Table 5-4 Jumper Settings for Error Recovery Procedures

Function	Install Jumper on Pins:				
	1/2	3/4	5/6	7/8	15/16
Restore factory defaults	0	0	0	0	1
Recover firmware	1	0	0	0	0
Note: 0 = No jumper installed; 1= Jumper installed					

Appendix A

Connectors

and Pinouts

This appendix describes the connectors and pinouts used on the UP2000+ Motherboard. Refer to Figure 2-1 in Chapter 2 for connector locations.

Note: *Pinouts of the Alpha Slot B Connectors J22 and J23 are described in the UP2000+ Technical Reference Manual, P/N 51-0039.*

A.1 Power Connector Pinouts

Pinouts for J26, the ATX power connector, are shown in Table A-1. J26 is a standard Molex 39-29-9202 connector.

Table A-1 ATX Power Connector Pinouts (J26)

Pin	Signal	Pin	Signal
1	+3.3 VDC	11	+3.3 VDC
2	+3.3 VDC	12	-12 VDC
3	GND	13	GND
4	+5 VDC	14	PS_ON
5	GND	15	GND
6	+5 VDC	16	GND
7	GND	17	GND
8	P_DCOK	18	-5 VDC
9	5V SB	19	+5 VDC
10	+12 VDC	20	+5 VDC

Table A-3 shows the pinouts for J28, an AUX ATX +3V power connector (PSCONN6). This part is a Molex 15-48-0412 connector.

Table A-2 AUX ATX Power Connector (PSCONN6) Pinouts (J28)

Pin	Signal	Pin	Signal
1	GND	4	+3.3 VDC
2	GND	5	+3.3 VDC
3	GND	6	+5 VDC

Pinouts for J25, the Alpha Slot B Module +12V power connector (PSCONN4) are shown in Table A-3. J25 is a Molex 15-24-4345 connector.

Table A-3 Alpha Slot B Module Power Connector (PSCONN4) Pinouts (J25)

Pin	Signal	Pin	Signal
1	+12 VDC	3	GND
2	GND	4	+5 VDC

A.2 Non-standard Connections

Some industry standard parts used in the UP2000+ are connected in non-standard applications. These connections are described in the following paragraphs.

A.2.1 Processor Configuration Connector Pinouts

The Processor configuration connector, J31, is a standard AMP 103239-2 connector. Table A-4 shows the pinouts for J31.

Table A-4 Processor Configuration Connector Pinout (J31)

Pin	Signal	Pin	Signal
1	N_113	2	VDDQ_ADJUST

A.2.2 AC Fail/Rack# Connector Pinouts

Table A-7 shows the pinouts for J36, the AC Fail/Rack# connector. This connector is a Molex 22-23-2031 part.

Table A-5 AC Fail/Rack# Connector Pinout (J36)

Pin	Signal	Pin	Signal
1	PSU_AC_Fail	3	Rack#_Enable
2	GND		

The AC Fail/Rack# connector, J36, provides two input signals to the UP2000+ Motherboard board, as follows:

- Pin 1 is the AC Fail signal from the power supply.
- Pin 2 is connected to Ground.
- Pin 3 is the signal RACK#, which sets a board configuration option to indicate the use of a PowerRAC Chassis 320 enclosure.

In a pedestal enclosure, J36 is not connected. A pull-up resistor on the UP2000+ Motherboard sets a signal (RACK#) value to 1, which indicates that the UP2000+ is not installed in a rackmount configuration. This signal is read by the firmware from the I^2C bus via the PCF8574 component (see the *UP2000+ Technical Reference Manual, P/N 51-0039*, for details on the I^2C logic).

In a PowerRAC Chassis 320 enclosure, a cable assembly from the enclosure power supply connects the power supply output signal ACFAIL to J36 pin 1, and connects J36 pin 3 to Ground. This sets the UP2000+ Motherboard signal RACK# value to 0, which indicates that the UP2000+ is installed in a rackmount configuration. Any other rackmount firmware options are controlled by the state read from the RACK# pin on the UP2000+ Motherboard.

UP2000+ firmware must know in which type of configuration the system is installed in order to properly interpret the system management inputs for voltage, fan presence and speed, and AC failure. There are different fans attached to the UP2000+ in the rackmount enclosure than are attached in the pedestal enclosure. There are also different power supplies for the two types of installation.

A.2.3 System Management Expansion Connector Pinout

The UP2000+ Motherboard System Management expansion connector is provided at J40. This connector is a standard Molex 87256-2011 part. Refer to Table A-6 for a complete pinout of this connector.

Table A-6 System Management Expansion Connector Pinout (J40)

Pin	Signal	Pin	Signal
1	Server_I2C_Data	11	HALT#
2	KEY	12	GND
3	Server_I2C_Clock	13	Reset#
4	GND	14	GND
5	SMI_INT#	15	NC
6	GND	16	GND
7	NC	17	SROMCLK_A
8	GND	18	SROMCLK_B
9	KBD_INH#	19	SROMDATA_A
10	GND	20	SROMDATA_B

A.2.4 Speaker Connector Pinouts

Table A-7 shows the pinouts for J41, the Speaker connector. This connector is an AMP 103239-4 part.

Table A-7 Speaker Connector Pinout (J41)

Pin	Signal	Pin	Signal
1	Spkr_signal	3	GND
2	+5V	4	+5V

A.2.5 Debug Port Connector Pinouts

The Alpha Slot B Module Debug Port connections are provided at J43 for the primary module, and J42 for the secondary module. Both connectors are standard AMP 103239-3 parts. Refer to Table A-6 for a complete pinout of these connectors.

Table A-8 Debug Port Connector Pinouts (J42, J43)

Pin	Signal	Pin	Signal
1	TEST_SROM_D_L	3	SROM_CLK_L
2	GND		

A.2.6 System Power Button Connector Pinouts

The System Power Button connector, J44, is a standard AMP 103239-2 connector. Table A-9 shows the pinouts for J44.

Table A-9 System Power Button Connector Pinout (J44)

Pin	Signal	Pin	Signal
1	Pwr_Button	2	GND

A.2.7 Halt Button Connector Pinouts

Table A-10 shows the pinouts for J45, the Halt Button connector. This connector is a standard Molex 22-23-2021 connector.

Table A-10 Halt Button Connector Pinout (J45)

Pin	Signal	Pin	Signal
1	GND	2	Halt_Button

A.2.8 Reset Button Connector Pinouts

The Reset Button connector, J47, is a standard Molex 22-23-2021 connector. Pinouts for J47 are shown in Table A-11.

Table A-11 Reset Button Connector Pinout (J47)

Pin	Signal	Pin	Signal
1	GND	2	Reset_But_L

A.3 Standard Connectors

Industry standard parts are used for most of the connections in the UP2000+. Refer to Table A-12 for a list of the connectors used and their functions.

Table A-12 UP2000+ Standard Connectors

Connector	Function	Part Number
J1	Power Good LED con.	Molex 22-23-2021
J2	HDD Activity LED con.	Molex 22-23-2021
J3	Keyboard and mouse	Foxconn MH11067-D2 or AMP 84405-1 or 84376-1
J4	USB	AMP 787617-1
J5	Parallel bus and COM1/COM2 serial line	Foxconn DM11351-Z5
J6	ISA expansion bus	AMP 645169-3
J7, J10	32-bit PCI bus	AMP 145154-4
J8, J9, J11, J12	64-bit PCI bus	AMP 1-145169-2
J13	Reset EPLD/ISP	Molex 87256-1011
J14–J21	SDRAM DIMMs	Berg 91159-61003
J24	FDD	Molex 87256-3411 or AMP 103308-7
J27	System Fail LED con.	Molex 22-23-2021
J29	Configuration jumper	Molex 10-89-9167
J30	IDE drive bus	Molex 87256-4011 or AMP 103308-8

Table A-12 UP2000+ Standard Connectors (Continued)

Connector	Function	Part Number
J32	Ultra-wide, Single-ended SCSI bus	Molex 15-87-0305
J33	Ultra2 LVDS / DIFF SCSI bus	Molex 15-87-0305
J34	Buzzer connector	Molex 22-23-2021
J35, J37–J39	Fan connectors	Molex 22-23-2031
J50, J51	Optional System Fan connectors	AMP 103239-2

Appendix B

Support,

Products and

Documentation

B.1 Customer Support

API NetWorks, Inc. provides assistance for their products on their web page at `www.api-networks.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.

B.2 Supporting Products

API NetWorks, Inc. maintains a Hardware Compatibility List on their website 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.api-networks.com` and check the Product Information list for Peripherals.

B.3 Alpha Products

API NetWorks, Inc. maintains information about other Alpha products on their website. Point your browser to `www.api-networks.com` and check the Product Information list for Alpha products.

B.4 Documentation

B.4.1 Alpha Documentation

Title	Vendor
Alpha Architecture Reference Manual, Third Edition	Compaq Computer Corporation, Digital Press order# EQ-W938E-DP
Alpha Architecture Handbook, Version 4	Compaq Computer Corporation Digital Press order# EC-QD2KC-TE
AlphaPC 264DP Technical Reference Manual	Compaq Computer Corporation, Digital Press order# EC-RBODA-TE
UP2000+ Quick Start Installation Guide (51-0041-1A)	API NetWorks, Inc.
UP2000+ Technical Reference Manual (51-0039-1A)	API NetWorks, Inc.
MACASE Chassis RFI Upgrade Kit Application Note (51-0038)	API NetWorks, Inc.

B.4.2 Third Party Documentation

You can order the following associated documentation directly from the vendor.

Title	Vendor
• *PCI Local Bus Specification, Revision 2.1* • *PCI Multimedia Design Guide, Revision 1.0* • *PCI System Design Guide* • *PCI-to-PCI Bridge Architecture Specification, Revision 0* • *PCI BIOS Specification, Revision 2.1*	PCI Special Interest Group U.S. 1-800-433-5177 International1-503-797-4207 FAX 1-503-234-6762
Computer Architecture	John L. Hennessy and David A. Patterson, Morgan Kaufman Publishers, San Mateo, CA, 1990

Index

Numerics

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

U

W

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:

API NetWorks, Inc.
130C Baker Street Extension
Concord, MA 01742
FAX: 978.371.317
E-mail: customer.support@api-networks.com

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: _____