



# AlphaPC 164LX Motherboard Tru64 UNIX

---

## User's Manual

Order Number: EC-R2ZPC-TE

**Revision/Update Information:** This is a revised document. It supersedes the *AlphaPC 164LX Motherboard DIGITAL UNIX User's Manual*, EC-R2ZPB-TE.

Compaq Computer Corporation

---

**March 1999**

The information in this publication is subject to change without notice.

COMPAQ COMPUTER CORPORATION SHALL NOT BE LIABLE FOR TECHNICAL OR EDITORIAL ERRORS OR OMISSIONS CONTAINED HEREIN, NOR FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES RESULTING FROM THE FURNISHING, PERFORMANCE, OR USE OF THIS MATERIAL. THIS INFORMATION IS PROVIDED "AS IS" AND COMPAQ COMPUTER CORPORATION DISCLAIMS ANY WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY AND EXPRESSLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR PARTICULAR PURPOSE, GOOD TITLE AND AGAINST INFRINGEMENT.

This publication contains information protected by copyright. No part of this publication may be photocopied or reproduced in any form without prior written consent from Compaq Computer Corporation.

© 1999 Digital Equipment Corporation  
All rights reserved. Printed in U.S.A.

The software described in this publication is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of the agreement.

COMPAQ, the Compaq logo, the Digital logo, and DIGITAL Registered in U.S. Patent and Trademark Office.

AlphaPC, DECnet, OpenVMS, and Tru64 are trademarks of Compaq Computer Corporation.

Microsoft, MS-DOS, Windows, and Windows NT are registered trademarks of Microsoft Corporation.

Intel is a registered trademark of Intel Corporation.

Other product names mentioned herein may be the trademarks of their respective companies.

---

# Contents

## 1 About This Manual

1.1	Manual Conventions and Terminology . . . . .	1-1
-----	--	-----

## 2 Features

2.1	Power Requirements . . . . .	2-4
2.2	Environmental Requirements . . . . .	2-4
2.3	Physical Parameters . . . . .	2-5
2.3.1	ATX Hole Specification . . . . .	2-6
2.3.2	ATX I/O Shield Requirements . . . . .	2-7

## 3 AlphaPC 164LX Jumper Configuration

3.1	CPU Speed Selection . . . . .	3-1
3.2	Bcache Size Jumpers (CF1 and CF2) . . . . .	3-1
3.3	Boot Option Jumper (CF7) . . . . .	3-3
3.4	Flash ROM Update Jumper (J28) . . . . .	3-3

## 4 AlphaPC 164LX Connector Pinouts

4.1	PCI Bus Connector Pinouts . . . . .	4-1
4.2	ISA Expansion Bus Connector Pinouts . . . . .	4-3
4.3	SDRAM DIMM Connector Pinouts . . . . .	4-4
4.4	EIDE Drive Bus Connector Pinouts . . . . .	4-5
4.5	Diskette (Floppy) Drive Bus Connector Pinouts . . . . .	4-6
4.6	Parallel Bus Connector Pinouts . . . . .	4-6
4.7	COM1/COM2 Serial Line Connector Pinouts . . . . .	4-7
4.8	Keyboard/Mouse Connector Pinouts . . . . .	4-7
4.9	SRAM Test Data Input Connector Pinouts . . . . .	4-8
4.10	Input Power Connector Pinouts . . . . .	4-8

4.11	Enclosure Fan Power Connector Pinouts . . . . .	4-8
4.12	Speaker Connector Pinouts . . . . .	4-9
4.13	Microprocessor Fan Power Connector Pinouts . . . . .	4-9
4.14	Power LED Connector Pinouts . . . . .	4-9
4.15	EIDE Drive LED Connector Pinouts . . . . .	4-10
4.16	Reset Button Connector Pinouts . . . . .	4-10
4.17	Halt Button Connector Pinouts . . . . .	4-10
4.18	Soft Power Connector Pinouts . . . . .	4-10

## 5 Memory and Microprocessor Configuration

5.1	Configuring SDRAM Memory . . . . .	5-1
5.2	Upgrading SDRAM Memory . . . . .	5-2
5.3	Increasing Microprocessor Speed . . . . .	5-3
5.3.1	Preparatory Information . . . . .	5-3
5.3.2	Required Tools . . . . .	5-3
5.3.3	Removing the 21164 Microprocessor . . . . .	5-4
5.3.4	Installing the 21164 Microprocessor . . . . .	5-4

## 6 Interrupts and ISA Bus Addresses

6.1	Interrupts . . . . .	6-1
6.2	ISA I/O Address Map . . . . .	6-2
6.2.1	Flash ROM Address Map . . . . .	6-3

## 7 Alpha SRM Console Firmware

7.1	Alpha SRM Console Firmware Conventions . . . . .	7-1
7.2	Basic Alpha SRM Console Command Descriptions . . . . .	7-3
7.3	Environment Variables for Alpha SRM Console Commands . . . . .	7-18
7.3.1	Environment Variable Descriptions . . . . .	7-18
7.3.1.1	Architecture-Required Environment Variables . . . . .	7-19
7.3.1.2	System-Defined Environment Variables . . . . .	7-20
7.4	Using the AlphaBIOS Firmware Update Utility to Update the Flash ROM . . . . .	7-22
7.4.1	AlphaBIOS Conventions . . . . .	7-22
7.4.2	Starting the AlphaBIOS . . . . .	7-23
7.4.3	Installing Alpha SRM Console Using AlphaBIOS Setup Program . . . . .	7-24
7.5	Installing the Tru64 UNIX Operating System . . . . .	7-29
7.5.1	Requirements . . . . .	7-29

## 8 Troubleshooting

8.1	Hardware Startup .....	8-1
8.2	Beep Codes .....	8-2
8.3	Post Codes .....	8-2
8.4	Fail-Safe Booter .....	8-4

## 9 Battery Recycle/Disposal Information

### A Support

A.1	Customer Support .....	A-1
A.2	Supporting Products .....	A-2
A.2.1	Memory .....	A-2
A.2.2	Thermal Products .....	A-2
A.2.3	Power Supply .....	A-2
A.2.4	Enclosure .....	A-3
A.3	Alpha Documentation .....	A-3
A.4	Third-Party Documentation .....	A-4

## Figures

2-1	AlphaPC 164LX Jumper/Connector/Component Location . . . . .	2-2
2-2	ATX Hole Specification . . . . .	2-6
2-3	ATX I/O Shield Dimensions . . . . .	2-7
3-1	AlphaPC 164LX Configuration Jumpers . . . . .	3-2
5-1	Fan/Heat-Sink Assembly . . . . .	5-5
7-1	AlphaBIOS Boot Screen . . . . .	7-23
7-2	AlphaBIOS Setup Screen . . . . .	7-24
7-3	AlphaBIOS Upgrade Options Screen . . . . .	7-25
7-4	AlphaBIOS Warning Screen . . . . .	7-26
7-5	AlphaBIOS Upgrade SRM Console Screen . . . . .	7-27
7-6	AlphaBIOS Upgrade Complete Screen . . . . .	7-28

## Tables

2-1	AlphaPC 164LX Features	2-1
2-2	AlphaPC 164LX Jumper/Connector/Component List	2-3
2-3	Power Supply DC Current Requirements	2-4
2-4	AlphaPC 164LX Motherboard Environmental Requirements	2-5
4-1	PCI Bus Connector Pinouts	4-1
4-2	ISA Expansion Bus Connector Pinouts (J30, J31)	4-3
4-3	SDRAM DIMM Connector Pinouts (J8 through J11)	4-4
4-4	EIDE Drive Bus Connector Pinouts (J6, J7)	4-5
4-5	Diskette (Floppy) Drive Bus Connector Pinouts (J15)	4-6
4-6	Parallel Bus Connector Pinouts (J13)	4-6
4-7	COM1/COM2 Serial Line Connector Pinouts (J4)	4-7
4-8	Keyboard/Mouse Connector Pinouts (J5)	4-7
4-9	SRAM Test Data Input Connector Pinouts (J29)	4-8
4-10	Input Power Connector Pinouts (J3)	4-8
4-11	Enclosure Fan (+12 V dc) Power Connector Pinouts (J2, J19)	4-8
4-12	Speaker Connector Pinouts (J20)	4-9
4-13	Microprocessor Fan Power Connector Pinouts (J18)	4-9
4-14	Power LED Connector Pinouts (J24)	4-9
4-15	EIDE Drive LED Connector Pinouts (J25)	4-10
4-16	Reset Button Connector Pinouts (J21)	4-10
4-17	Halt Button Connector Pinouts (J22)	4-10
4-18	Soft Power Connector Pinouts (J1)	4-10
5-1	AlphaPC 164LX SDRAM Memory Configurations	5-2
6-1	ISA Interrupts	6-1
6-2	ISA I/O Address Map	6-2
7-1	AlphaBIOS Keys	7-22
8-1	Beep Codes	8-2
8-2	Post Codes	8-2





---

# About This Manual

This manual describes the AlphaPC 164LX motherboard, a module for computing systems based on the Alpha 21164 microprocessor and the companion 21174 core logic chip. It describes the motherboard's features and how to set its configuration jumpers. This manual helps users to install and populate the AlphaPC 164LX motherboard with memory modules and peripheral cards.

## 1.1 Manual Conventions and Terminology

The following conventions are used in this manual.

**Caution:** Cautions indicate potential damage to equipment, software, or data.

**Note:** Notes provide additional information about a topic.

**Numbering:** All numbers are decimal or hexadecimal unless otherwise indicated. In case of ambiguity, a subscript indicates the radix of nondecimal numbers. For example, 19 is a decimal number, but  $19_{16}$  and 19A are hexadecimal numbers.

**Extents:** Extents are specified by a single number or a pair of numbers in square brackets ([ ]) separated by a colon (:), and are inclusive. For example, bits [7:3] specify an extent including bits 7, 6, 5, 4, and 3. Multiple bit fields are shown as extents.

**Register Figures:** Register figures have bit and field position numbering starting at the right (low-order) and increasing to the left (high-order).

**Signal Names:** All signal names are printed in **boldface** type. Signal names that originate in an industry-standard specification, such as PCI or IDE, are printed in the case as found in the specification (usually uppercase). Active low signals have either a pound sign “#” appended, or a “not” overscore bar; for example, **DEVSEL#** and **RESET**.

## Manual Conventions and Terminology

**Italic Type:** *Italic* type emphasizes important information and indicates complete titles of documents.

**Terms:** The following terms are used in this manual:

---

<b>This term...</b>	<b>Refers to...</b>
Alpha SRM console	The Alpha SRM Console firmware
DIGITAL UNIX installation guide	The <i>DIGITAL UNIX Installation Guide</i>
Tru64 UNIX	The Tru64 UNIX (formerly DIGITAL UNIX) operating system

---

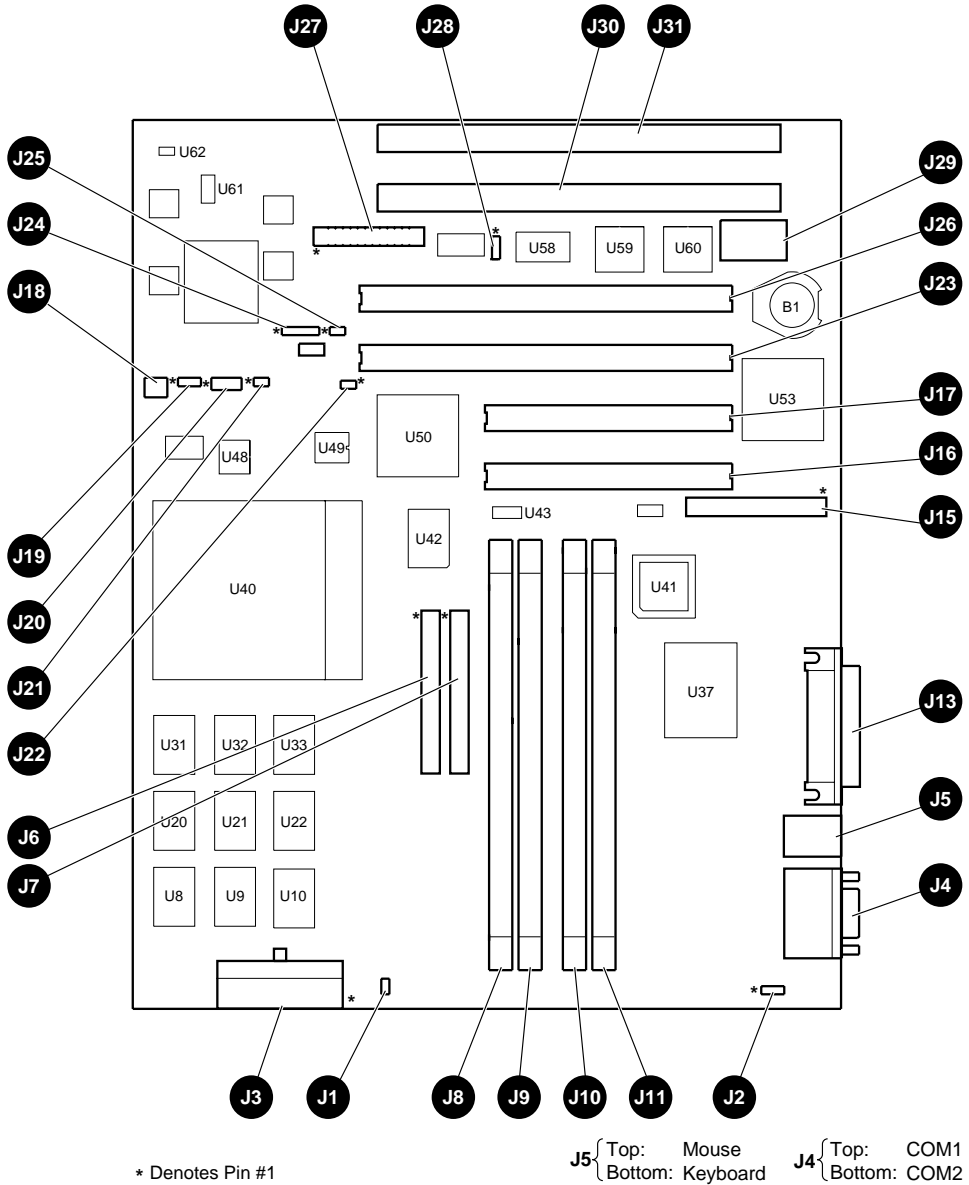
Table 2–1 provides an overview of the AlphaPC 164LX motherboard’s features.

**Table 2–1 AlphaPC 164LX Features**

Feature	Description
Microprocessor	Alpha 21164 microprocessor (64-bit RISC)
Core logic chip	21174 core logic chip, comprising a single control chip that provides an interface to system memory and the PCI bus
Synchronous DRAM (SDRAM) memory	32MB to 1GB memory array — Two banks of 128-bit memory; 168-pin unbuffered SDRAM DIMMs
Caching:	
L1 Icache	8KB, direct-mapped instruction cache on the CPU chip
L1 Dcache	8KB, direct-mapped data cache on the CPU chip
L2 Scache	96KB, three-way, set-associative, write-back unified instruction and data cache on the CPU chip
L3 backup cache	Onboard 2MB, direct-mapped, synchronous SSRAM backup cache with 128-bit data path
I/O and miscellaneous support	32-bit and 64-bit, 33-MHz PCI Two 64-bit and two 32-bit PCI expansion slots PCI/EIDE control (CMD PCI0646) Intel 82378ZB PCI-to-ISA bridge chip Two dedicated ISA expansion slots SMC FDC37C935 combination controller chip provides control for diskettes, two UARTs with modem control, parallel port, keyboard, mouse, and time-of-year clock 1MB flash ROM
Firmware	Alpha SRM Console firmware

Figure 2–1 shows the AlphaPC 164LX motherboard and its components and Table 2–2 describes these components.

**Figure 2–1 AlphaPC 164LX Jumper/Connector/Component Location**



FM-05933.A14

**Table 2–2 AlphaPC 164LX Jumper/Connector/Component List**

<b>Item No.</b>	<b>Description</b>	<b>Item No.</b>	<b>Description</b>
B1	RTC battery (CR2032)	J25	Hard-drive LED connector
J1	Soft power connector	J26	PCI slot 0 (64-bit)
J2	Fan power, enclosure (+12V)	J27	System configuration jumpers
J3	Power (+3V, +5V, -5V, +12V, -12V)	J28	Flash ROM update enable/disable jumper
J4	COM1/COM2 (DB9) connectors	J29	SRAM test port connector
J5	Keyboard/mouse connectors	J30	ISA slot 1
J6	EIDE drive 2/3 connector	J31	ISA slot 0
J7	EIDE drive 0/1 connector	U8 to U10	Cache SRAM (L3)
J8	SDRAM DIMM 0 [0:63] connector	U20 to U22	Cache SRAM (L3)
J9	SDRAM DIMM 1 [64:128] connector	U31 to U33	Cache SRAM (L3)
J10	SDRAM DIMM 2 [0:63] connector	U37	I/O interface and address control (DS 21174-CA)
J11	SDRAM DIMM 3 [64:128] connector	U40	Microprocessor, socket (DS Alpha 21164)
J12	Reserved	U41	Patch 8K PAL
J13	Parallel I/O connector	U42	EIDE controller
J14	Reserved	U43	System clock PLL (CY2308)
J15	Diskette (floppy) drive connector	U48	Microprocessor clock synthesizer (MC12439)
J16	PCI slot 3 (32-bit)	U49	Serial ROM, socketed (Xilinx XC17128D)
J17	PCI slot 2 (32-bit)	U50	PCI-to-ISA bridge (Intel 82378ZB)
J18	Microprocessor fan/fan sense connector	U53	Combination controller, Super I/O (SMC FDC37C935)
J19	Enclosure fan +12V power connector	U58	Flash ROM (1MB)
J20	Speaker connector	U59	PCI arbiter PAL
J21	Reset button connector	U60	PCI interrupt request PAL
J22	Halt button connector	U61	Power controller
J23	PCI slot 1 (64-bit)	U62	Power sense
J24	Power LED connector		

# Power Requirements

## 2.1 Power Requirements

The AlphaPC 164LX motherboard has a total power dissipation of 100 W, excluding any plug-in PCI and ISA devices. Table 2–3 lists the power requirement for each dc supply voltage.

The power supply must be ATX-compliant.

**Table 2–3 Power Supply DC Current Requirements**

Voltage/Tolerance	Current <sup>1</sup>
+3.3 V dc, ±5%	5.0 A
+5 V dc, ±5%	14.0 A
–5 V dc, ±5%	0 A
+12 V dc, ±5%	1.0 A
–12 V dc, ±5%	100.0 mA

<sup>1</sup> Values indicated are for an AlphaPC 164LX motherboard with an Alpha 21164 microprocessor operating at 600 MHz, with 64MB SDRAM, excluding adapter cards and disk drives.

**Caution:** **Fan sensor required.** The 21164 microprocessor cooling fan *must* have a built-in sensor that will drive a signal if the airflow stops. The sensor is connected to motherboard connector J18. When the signal is generated, it resets the system.

## 2.2 Environmental Requirements

The 21164 microprocessor is cooled by a small fan blowing directly into the chip's heat sink. The AlphaPC 164LX motherboard is designed to run efficiently by using only this fan. Additional fans may be necessary depending upon cabinetry and the requirements of plug-in cards.

The AlphaPC 164LX motherboard is specified to run within the environment listed in Table 2–4.

**Table 2–4 AlphaPC 164LX Motherboard Environmental Requirements**

Parameter	Specification
Operating temperature	10°C to 40°C (50°F to 104°F)
Storage temperature	–55°C to 125°C (–67°F to 257°F)
Relative humidity	10% to 90% with maximum wet bulb temperature 28°C (82°F) and minimum dew point 2°C (36°F)
Rate of (dry bulb) temperature change	11°C/hour $\pm$ 2°C/hour (20°F/hour $\pm$ 4°F/hour)

### 2.3 Physical Parameters

The AlphaPC 164LX motherboard is an ATX-size printed-wiring board (PWB) with the following dimensions:

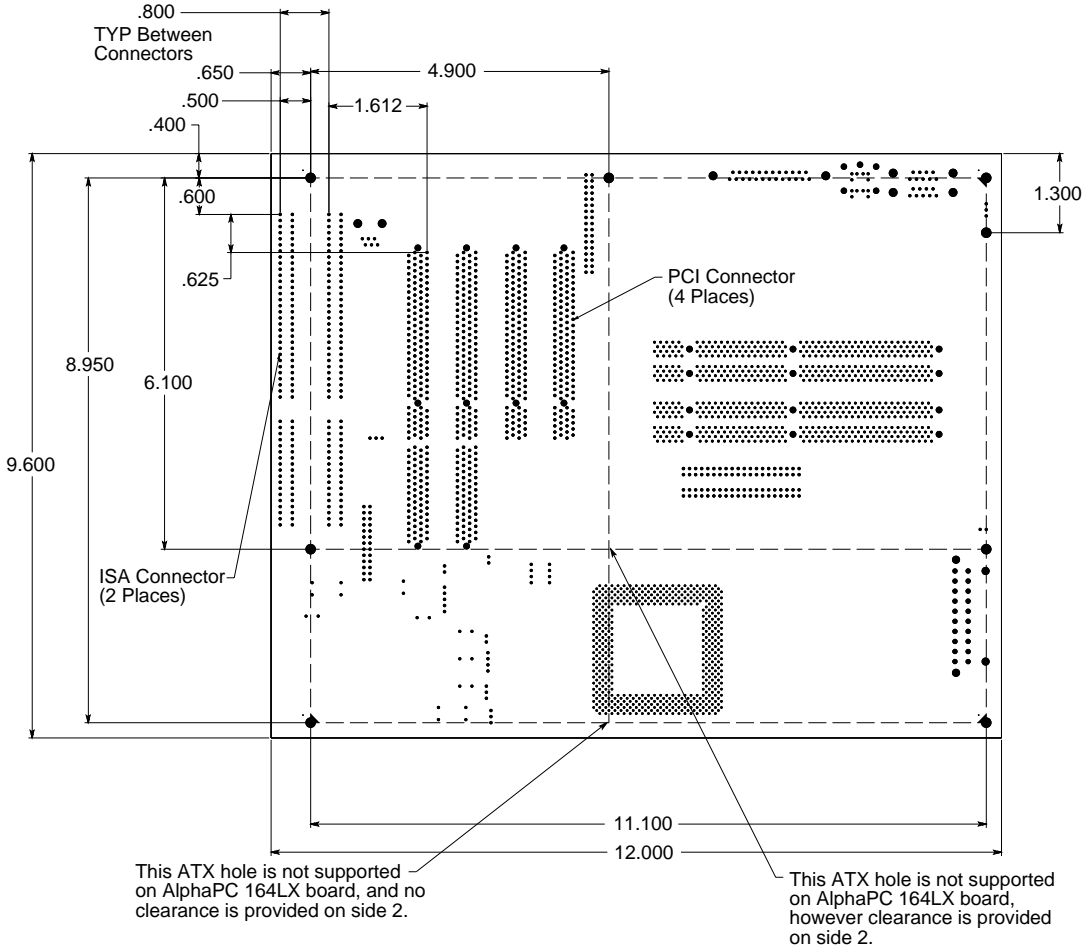
- Length: 30.48 cm (12.0 in  $\pm$ 0.0005 in)
- Width: 24.38 cm (9.6 in  $\pm$ 0.0005 in)
- Height: 6.86 cm (2.7 in)

# Physical Parameters

## 2.3.1 ATX Hole Specification

Figure 2-2 shows the ATX Hole Specification for the AlphaPC 164LX. Measurements are shown in inches.

Figure 2-2 ATX Hole Specification



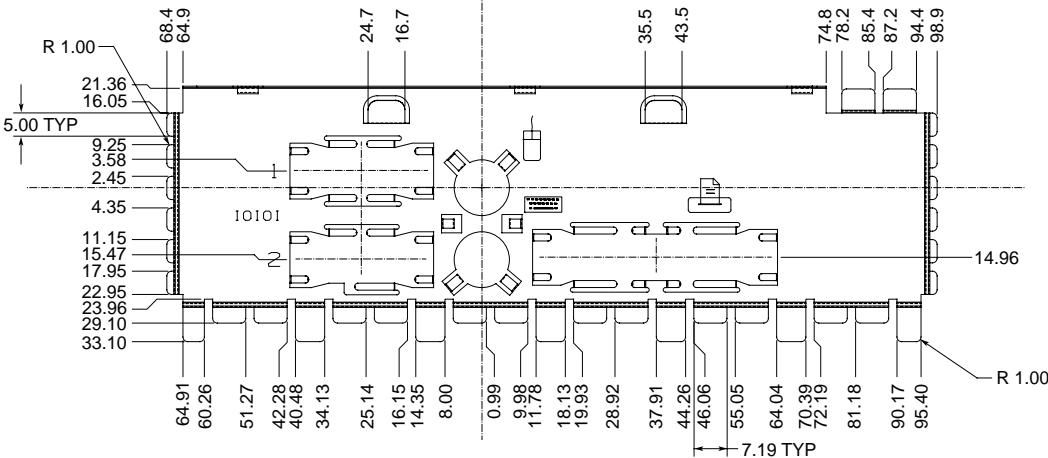
FM-06012.A14



2.3.2 ATX I/O Shield Requirements

Figure 2-3 shows the ATX I/O shield dimensions for the AlphaPC 164LX. Measurements are shown in millimeters.

Figure 2-3 ATX I/O Shield Dimensions



FM-05986.A14



---

## AlphaPC 164LX Jumper Configuration

The AlphaPC 164LX motherboard has two groups of jumpers located at J27 and J28, as shown in Figure 2–1. These jumpers set the hardware configuration and boot options. Figure 3–1 shows these jumper configurations.

### 3.1 CPU Speed Selection

The clock synthesizer at location U47 makes it possible to change the frequency of the microprocessor's system clock output without having to change the clock crystal. Simply set the system clock divisor jumpers to adjust the frequency of the microprocessor's system clock output. These system clock divisor jumpers are located at J27–1/2 (IRQ3), J27–3/4 (IRQ2), J27–5/6 (IRQ1), and J27–7/8 (IRQ0). The jumper configuration is set in IRQ3 through IRQ0. These four jumpers set the speed at power-up as listed in Figure 3–1. The microprocessor frequency divided by the ratio determines the system clock frequency.

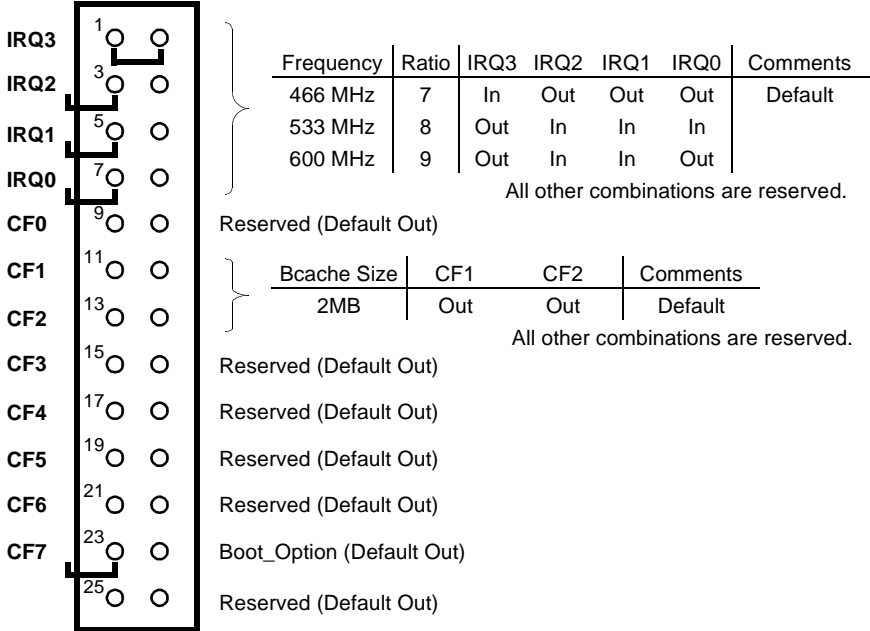
### 3.2 Bcache Size Jumpers (CF1 and CF2)

The Bcache size jumpers are located at J27–11/12 (CF) and J27–13/14 (CF2). These jumpers configure the Bcache as specified in Figure 3–1.

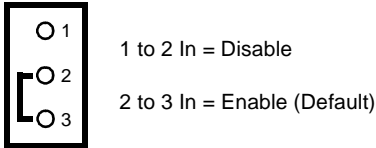
# Bcache Size Jumpers (CF1 and CF2)

Figure 3–1 AlphaPC 164LX Configuration Jumpers

## J27 System Configuration Jumpers



## J28 Flash ROM Update Jumpers



### 3.3 Boot Option Jumper (CF7)

The boot option jumper is located at J27–23/24 (CF7). The default position for this jumper is out (Figure 3–1). This jumper selects the image to be loaded into memory from the system flash ROM. With the jumper out, the SRM Console firmware is loaded. With the jumper in, the fail-safe booter is loaded. For more information about the fail-safe booter, refer to Section 8.4.

### 3.4 Flash ROM Update Jumper (J28)

When J28–2/3 are jumpered together (default), the flash ROM is write-enabled. When J28–1/2 are jumpered together, the flash ROM is write-protected.



# 4

## AlphaPC 164LX Connector Pinouts

This chapter contains pinouts for all of the AlphaPC 164LX connectors. See Figure 2–1 for connector locations.

### 4.1 PCI Bus Connector Pinouts

Table 4–1 shows the PCI bus connector pinouts.

**Table 4–1 PCI Bus Connector Pinouts**

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
<b>32-Bit and 64-Bit PCI Connectors (J16, J17, J23, J26)</b>							
A1	TRST#	A2	+12V	A3	TMS	A4	TDI
A5	Vdd	A6	INTA	A7	INTC	A8	Vdd
A9	—	A10	Vdd	A11	—	A12	Gnd
A13	Gnd	A14	—	A15	RST#	A16	Vdd
A17	GNT#	A18	Gnd	A19	—	A20	AD[30]
A21	+3V	A22	AD[28]	A23	AD[26]	A24	Gnd
A25	AD[24]	A26	IDSEL	A27	+3V	A28	AD[22]
A29	AD[20]	A30	Gnd	A31	AD[18]	A32	AD[16]
A33	+3V	A34	FRAME#	A35	Gnd	A36	TRDY#
A37	STOP#	A38	STOP#	A39	+3V	A40	SDONE
A41	SBO#	A42	Gnd	A43	PAR	A44	AD[15]
A45	+3V	A46	AD[13]	A47	AD[11]	A48	Gnd
A49	AD[09]	A50	Not used	A51	Not used	A52	C/BE#[0]
A53	+3V	A54	AD[06]	A55	AD[04]	A56	Gnd
A57	AD[02]	A58	AD[00]	A59	Vdd	A60	REQ64#
A61	Vdd	A62	Vdd	B1	-12V	B2	TCK
B3	Gnd	B4	TDO	B5	Vdd	B6	Vdd
B7	INTB	B8	INTD	B9	PRSENT1#	B10	—

## PCI Bus Connector Pinouts

**Table 4–1 PCI Bus Connector Pinouts (Continued)**

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
B11	PRSENT2#	B12	Gnd	B13	Gnd	B14	—
B15	Gnd	B16	CLK	B17	Gnd	B18	REQ#
B19	Vdd	B20	AD[31]	B21	AD[29]	B22	Gnd
B23	AD[27]	B24	AD[25]	B25	+3V	B26	C/BE#[3]
B27	AD[23]	B28	Gnd	B29	AD[21]	B30	AD[19]
B31	+3V	B32	AD[17]	B33	C/BE#[2]	B34	Gnd
B35	IRDY#	B36	+3V	B37	DEVSEL#	B38	Gnd
B39	LOCK#	B40	PERR#	B41	+3V	B42	SERR#
B43	+3V	B44	C/BE#[1]	B45	AD[14]	B46	Gnd
B47	AD[12]	B48	AD[10]	B49	Gnd	B50	Not used
B51	Not used	B52	AD[08]	B53	AD[07]	B54	+3V
B55	AD[05]	B56	AD[03]	B57	Gnd	B58	AD[01]
B59	Vdd	B60	ACK64#	B61	Vdd	B62	Vdd
<b>64-Bit PCI Connectors Only (J23, J26)</b>							
A63	Gnd	A64	C/BE#[7]	A65	C/BE#[5]	A66	Vdd
A67	PAR64	A68	D[62]	A69	Gnd	A70	D[60]
A71	D[58]	A72	Gnd	A73	D[56]	A74	D[54]
A75	Vdd	A76	D[52]	A77	D[50]	A78	Gnd
A79	D[48]	A80	D[46]	A81	Gnd	A82	D[44]
A83	D[42]	A84	Vdd	A85	D[40]	A86	D[38]
A87	Gnd	A88	D[36]	A89	D[34]	A90	Gnd
A91	D[32]	A92	—	A93	Gnd	A94	—
B63	—	B64	Gnd	B65	C/BE#[6]	B66	C/BE#[4]
B67	Gnd	B68	D[63]	B69	D[61]	B70	Vdd
B71	D[59]	B72	D[57]	B73	Gnd	B74	D[55]
B75	D[53]	B76	Gnd	B77	D[51]	B78	D[49]
B79	Vdd	B80	D[47]	B81	D[45]	B82	Gnd
B83	D[43]	B84	D[41]	B85	Gnd	B86	D[39]
B87	D[37]	B88	Vdd	B89	D[35]	B90	D[33]
B91	Gnd	B92	—	B93	—	B94	Gnd



## 4.2 ISA Expansion Bus Connector Pinouts

Table 4–2 shows the ISA expansion bus connector pinouts.

**Table 4–2 ISA Expansion Bus Connector Pinouts (J30, J31)**

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	Gnd	2	IOCHCK#	3	RSTDRV	4	SD7
5	Vdd	6	SD6	7	IRQ9	8	SD5
9	–5V	10	SD4	11	DRQ2	12	SD3
13	–12V	14	SD2	15	ZEROWS#	16	SD1
17	+12V	18	SD0	19	Gnd	20	IOCHRDY
21	SMEMW#	22	AEN	23	SMEMR#	24	SA19
25	IOW#	26	SA18	27	IOR#	28	SA17
29	DACK3#	30	SA16	31	DRQ3	32	SA15
33	DACK1#	34	SA14	35	DRQ1	36	SA13
37	REFRESH#	38	SA12	39	SYCLK	40	SA11
41	IRQ7	42	SA10	43	IRQ6	44	SA9
45	IRQ5	46	SA8	47	IRQ4	48	SA7
49	IRQ3	50	SA6	51	DACK2#	52	SA5
53	TC	54	SA4	55	BALE	56	SA3
57	Vdd	58	SA2	59	OSC	60	SA1
61	Gnd	62	SA0	63	MEMCS16#	64	SBHE#
65	IOCS16#	66	LA23	67	IRQ10	68	LA22
69	IRQ11	70	LA21	71	IRQ12	72	LA20
73	IRQ15	74	LA19	75	IRQ14	76	LA18
77	DACK0#	78	LA17	79	DRQ0	80	MEMR#
81	DACK5#	82	MEMW#	83	DRQ5	84	SD8
85	DACK6#	86	SD9	87	DRQ6	88	SD10
89	DACK7#	90	SD11	91	DRQ7	92	SD12
93	Vdd	94	SD13	95	MASTER#	96	SD14
97	Gnd	98	SD15	—	—	—	—

## SDRAM DIMM Connector Pinouts

### 4.3 SDRAM DIMM Connector Pinouts

Table 4–3 shows the SDRAM DIMM connector pinouts.

**Table 4–3 SDRAM DIMM Connector Pinouts (J8 through J11)<sup>1</sup>**

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	Gnd	2	DQ0	3	DQ1	4	DQ2
5	DQ3	6	+3V	7	DQ4	8	DQ5
9	DQ6	10	DQ7	11	DQ8	12	Gnd
13	DQ9	14	DQ10	15	DQ11	16	DQ12
17	DQ13	18	+3V	19	DQ14	20	DQ15
21	CB0	22	CB1	23	Gnd	24	NC
25	NC	26	+3V	27	$\overline{\text{WE}}$	28	DQMB0
29	DQMB1	30	$\overline{\text{S0}}$	31	NC	32	Gnd
33	A0	34	A2	35	A4	36	A6
37	A8	38	A10	39	A12	40	+3V
41	+3V	42	CK0	43	Gnd	44	NC
45	$\overline{\text{S2}}$	46	DQMB2	47	DQMB3	48	NC
49	+3V	50	NC	51	NC	52	CB2
53	CB3	54	Gnd	55	DQ16	56	DQ17
57	DQ18	58	DQ19	59	+3V	60	DQ20
61	NC	62	NC	63	CKE1	64	Gnd
65	DQ21	66	DQ22	67	DQ23	68	Gnd
69	DQ24	70	DQ25	71	DQ26	72	DQ27
73	+3V	74	DQ28	75	DQ29	76	DQ30
77	DQ31	78	Gnd	79	CK2	80	NC
81	NC	82	SDA	83	SCL	84	+3V
85	Gnd	86	DQ32	87	DQ33	88	DQ34
89	DQ35	90	+3V	91	DQ36	92	DQ37
93	DQ38	94	DQ39	95	DQ40	96	Gnd
97	DQ41	98	DQ42	99	DQ43	100	DQ44
101	DQ45	102	+3V	103	DQ46	104	DQ47
105	CB4	106	CB5	107	Gnd	108	NC
109	NC	110	+3V	111	$\overline{\text{CAS}}$	112	DQMB4
113	DQMB5	114	S1	115	$\overline{\text{RAS}}$	116	Gnd
117	A1	118	A3	119	A5	120	A7
121	A9	122	BA0	123	A13	124	+3V

## EIDE Drive Bus Connector Pinouts

**Table 4–3 SDRAM DIMM Connector Pinouts (J8 through J11)<sup>1</sup> (Continued)**

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
125	<b>CK1</b>	126	<b>BA1<sup>2</sup></b>	127	<b>Gnd</b>	128	<b>CKE0</b>
129	<b><u>S3</u></b>	130	<b>DQMB6</b>	131	<b>DQMB7</b>	132	<b>PD<sup>3</sup></b>
133	<b>+3V</b>	134	<b>NC</b>	135	<b>NC</b>	136	<b>CB6</b>
137	<b>CB7</b>	138	<b>Gnd</b>	139	<b>DQ48</b>	140	<b>DQ49</b>
141	<b>DQ50</b>	142	<b>DQ51</b>	143	<b>+3V</b>	144	<b>DQ52</b>
145	<b>NC</b>	146	<b>NC</b>	147	<b>PD</b>	148	<b>Gnd</b>
149	<b>DQ53</b>	150	<b>DQ54</b>	151	<b>DQ55</b>	152	<b>Gnd</b>
153	<b>DQ56</b>	154	<b>DQ57</b>	155	<b>DQ58</b>	156	<b>DQ59</b>
157	<b>+3V</b>	158	<b>DQ60</b>	159	<b>DQ61</b>	160	<b>DQ62</b>
161	<b>DQ63</b>	162	<b>Gnd</b>	163	<b>CK3</b>	164	<b>NC</b>
165	<b>SA0</b>	166	<b>SA1</b>	167	<b>SA2</b>	168	<b>+3V</b>

<sup>1</sup> Pins 1 through 84 are on the front side and pins 85 through 168 are on the back side.

<sup>2</sup> The AlphaPC 164LX uses **BA1** as both **BA1** and **ADDR12**. Therefore, four-bank DIMMs using **ADDR[11:0]** are the maximum size. (Two-bank DIMMs can use **ADDR[12:0]**.)

<sup>3</sup> Pull-down.

## 4.4 EIDE Drive Bus Connector Pinouts

Table 4–4 shows the EIDE drive bus connector pinouts.

**Table 4–4 EIDE Drive Bus Connector Pinouts (J6, J7)**

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	<b><u>RESET</u></b>	2	<b>Gnd</b>	3	<b>IDE_D7</b>	4	<b>IDE_D8</b>
5	<b>IDE_D6</b>	6	<b>IDE_D9</b>	7	<b>IDE_D5</b>	8	<b>IDE_D10</b>
9	<b>IDE_D4</b>	10	<b>IDE_D11</b>	11	<b>IDE_D3</b>	12	<b>IDE_D12</b>
13	<b>IDE_D2</b>	14	<b>IDE_D13</b>	15	<b>IDE_D1</b>	16	<b>IDE_D14</b>
17	<b>IDE_D0</b>	18	<b>IDE_D15</b>	19	<b><u>Gnd</u></b>	20	<b>NC (key pin)</b>
21	<b><u>MARQ</u></b>	22	<b>Gnd</b>	23	<b><u>IOW</u></b>	24	<b>Gnd</b>
25	<b><u>IOR</u></b>	26	<b>Gnd</b>	27	<b>CHRDY</b>	28	<b><u>BALE</u></b>
29	<b>MACK</b>	30	<b>Gnd</b>	31	<b>IRQ</b>	32	<b><u>IOCS16</u></b>
33	<b><u>ADDR1</u></b>	34	<b><u>NC</u></b>	35	<b><u>ADDR0</u></b>	36	<b>ADDR2</b>
37	<b><u>CS0</u></b>	38	<b><u>CS1</u></b>	39	<b><u>ACT</u></b>	40	<b>Gnd</b>

## Diskette (Floppy) Drive Bus Connector Pinouts

### 4.5 Diskette (Floppy) Drive Bus Connector Pinouts

Table 4–5 shows the diskette (floppy) drive bus connector pinouts.

**Table 4–5 Diskette (Floppy) Drive Bus Connector Pinouts (J15)**

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	Gnd	2	DEN0	3	Gnd	4	NC
5	Gnd	6	DEN1	7	Gnd	8	INDEX
9	Gnd	10	MTR0	11	Gnd	12	DR1
13	Gnd	14	DR0	15	Gnd	16	MTR1
17	Gnd	18	DIR	19	Gnd	20	STEP
21	Gnd	22	WDATA	23	Gnd	24	WGATE
25	Gnd	26	TRK0	27	Gnd	28	WRTPRT
29	ID0	30	RDATA	31	Gnd	32	HDSEL
33	ID1	34	DSKCHG	—	—	—	—

### 4.6 Parallel Bus Connector Pinouts

Table 4–6 shows the parallel bus connector pinouts.

**Table 4–6 Parallel Bus Connector Pinouts (J13)**

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	STB	2	PD0	3	PD1	4	PD2
5	PD3	6	PD4	7	PD5	8	PD6
9	PD7	10	ACK	11	BUSY	12	PE
13	SLCT	14	AFD	15	ERR	16	INIT
17	SLIN	18	Gnd	19	Gnd	20	Gnd
21	Gnd	22	Gnd	23	Gnd	24	Gnd
25	Gnd	—	—	—	—	—	—

## 4.7 COM1/COM2 Serial Line Connector Pinouts

Table 4–7 shows the COM1/COM2 serial line connector pinouts.

**Table 4–7 COM1/COM2 Serial Line Connector Pinouts (J4)**

COM1 Pin (Top)	COM1 Signal	COM2 Pin (Bottom)	COM2 Signal
1	<b>DCD1</b>	1	<b>DCD2</b>
2	<b>RxD1</b>	2	<b>RxD2</b>
3	<b>TxD1</b>	3	<b>TxD2</b>
4	<b>DTR1</b>	4	<b>DTR2</b>
5	<b>SG1</b>	5	<b>SG2</b>
6	<b>DSR1</b>	6	<b>DSR2</b>
7	<b>RTS1</b>	7	<b>RTS2</b>
8	<b>CTS1</b>	8	<b>CTS2</b>
9	<b>RI1</b>	9	<b>RI2</b>

## 4.8 Keyboard/Mouse Connector Pinouts

Table 4–8 shows the keyboard/mouse connector pinouts.

**Table 4–8 Keyboard/Mouse Connector Pinouts (J5)**

Keyboard Pin (Top)	Keyboard Signal	Mouse Pin (Bottom)	Mouse Signal
1	<b>KBDATA</b>	1	<b>MSDATA</b>
2	<b>NC</b>	2	<b>NC</b>
3	<b>Gnd</b>	3	<b>Gnd</b>
4	<b>Vdd</b>	4	<b>Vdd</b>
5	<b>KBCLK</b>	5	<b>MSCLK</b>
6	<b>NC</b>	6	<b>NC</b>

# SROM Test Data Input Connector Pinouts

## 4.9 SROM Test Data Input Connector Pinouts

Table 4–9 shows the SROM test data input connector pinouts.

**Table 4–9 SROM Test Data Input Connector Pinouts (J29)**

Pin	Signal	Name
1	NC	—
2	<b>SROM_CLK_L</b>	Clock out
3	<b>Gnd</b>	—
4	NC	—
5	<b>TEST_SROM_D_L</b>	SROM serial data in
6	NC	—

## 4.10 Input Power Connector Pinouts

Table 4–10 shows the input power connector pinouts.

**Table 4–10 Input Power Connector Pinouts (J3)<sup>1</sup>**

Pin	Voltage	Pin	Voltage	Pin	Voltage	Pin	Voltage
1	+3.3 V dc	2	+3.3 V dc	3	<b>Gnd</b>	4	+5 V dc
5	<b>Gnd</b>	6	+5 V dc	7	<b>Gnd</b>	8	<b>P_DCOK</b>
9	5 V SB	10	+12 V dc	11	+3.3 V dc	12	–12 V dc
13	<b>Gnd</b>	14	<b>PS_ON</b>	15	<b>Gnd</b>	16	<b>Gnd</b>
17	<b>Gnd</b>	18	–5 V dc	19	+5 V dc	20	+5 V dc

<sup>1</sup> This pinout is ATX-compliant.

## 4.11 Enclosure Fan Power Connector Pinouts

Table 4–11 shows the enclosure fan power connector pinouts.

**Table 4–11 Enclosure Fan (+12 V dc) Power Connector Pinouts (J2, J19)**

Pin	Voltage
1	<b>Gnd</b>
2	+12 V dc
3	<b>Gnd</b>

## 4.12 Speaker Connector Pinouts

Table 4–12 shows the speaker connector pinouts.

**Table 4–12 Speaker Connector Pinouts (J20)**

Pin	Signal	Name
1	SPKR	Speaker output
2	Gnd	—
3	Gnd	—
4	Gnd	—

## 4.13 Microprocessor Fan Power Connector Pinouts

Table 4–13 shows the microprocessor fan power connector pinouts.

**Table 4–13 Microprocessor Fan Power Connector Pinouts (J18)**

Pin	Signal	Name
1	+12 V dc	—
2	FAN_CONN_L	Fan connected
3	Gnd	—

## 4.14 Power LED Connector Pinouts

Table 4–14 shows the power LED connector pinouts.

**Table 4–14 Power LED Connector Pinouts (J24)**

Pin	Signal	Name
1	POWER_LED_L	Power LED input
2	Gnd	—
3	NC	—
4	NC	—
5	NC	—

# EIDE Drive LED Connector Pinouts

## 4.15 EIDE Drive LED Connector Pinouts

Table 4–15 shows the EIDE drive LED connector pinouts.

**Table 4–15 EIDE Drive LED Connector Pinouts (J25)**

Pin	Signal	Name
1	HD_ACT_L	Hard drive active
2	HD_LED_L	Hard drive LED input

## 4.16 Reset Button Connector Pinouts

Table 4–16 shows the reset button connector pinouts.

**Table 4–16 Reset Button Connector Pinouts (J21)**

Pin	Signal	Name
1	RESET_BUTTON	Reset system
2	Gnd	—

## 4.17 Halt Button Connector Pinouts

Table 4–17 shows the halt button connector pinouts.

**Table 4–17 Halt Button Connector Pinouts (J22)**

Pin	Signal	Name
1	HALT_BUTTON	Halt system
2	Gnd	—

**Note:** The Halt button is not used with the Windows NT operating system.

## 4.18 Soft Power Connector Pinouts

Table 4–18 shows the soft power connector pinouts.

**Table 4–18 Soft Power Connector Pinouts (J1)**

Pin	Signal	Name
1	Input	System power on/off
2	Gnd	—



---

## Memory and Microprocessor Configuration

For higher system speed or greater throughput, you can upgrade SDRAM memory by replacing DIMMs with those of greater size.

When configuring or upgrading SDRAM, observe the following rules:

- Each DIMM must be a 168-pin unbuffered version and have a frequency of 100 MHz.
- All DIMMs must be of equal size if they are in the same bank.

### 5.1 Configuring SDRAM Memory

Although not an exhaustive list, Table 5–1 lists the tested SDRAM memory configurations available.

For a list of vendors who supply components and accessories for the AlphaPC 164LX, see Appendix A.

Refer to Figure 2–1 for DIMM connector locations.

**Note:** 1Mb × 72 is not supported.

## Upgrading SDRAM Memory

**Table 5–1 AlphaPC 164LX SDRAM Memory Configurations**

Total Memory	Bank 0		Bank 1	
	J8	J9	J10	J11
32MB	2Mb x 72	2Mb x 72	—	—
64MB	2Mb x 72	2Mb x 72	2Mb x 72	2Mb x 72
	4Mb x 72	4Mb x 72	—	—
96MB	4Mb x 72	4Mb x 72	2Mb x 72	2Mb x 72
128MB	4Mb x 72	4Mb x 72	4Mb x 72	4Mb x 72
	8Mb x 72	8Mb x 72	—	—
160MB	8Mb x 72	8Mb x 72	2Mb x 72	2Mb x 72
192MB	8Mb x 72	8Mb x 72	4Mb x 72	4Mb x 72
256MB	8Mb x 72	8Mb x 72	8Mb x 72	8Mb x 72
	16Mb x 72	16Mb x 72	—	—
512MB	16Mb x 72	16Mb x 72	16Mb x 72	16Mb x 72

## 5.2 Upgrading SDRAM Memory

You can upgrade memory in the AlphaPC 164LX by adding more DIMMs or replacing the ones that you have with a greater size. Refer to Figure 2–1 for DIMM connector locations.

Use the following general guidelines:

1. *Observe antistatic precautions.* Handle DIMMs only at the edges to prevent damage.
2. Remove power from the system.
3. Open levers and align the DIMM.
4. Firmly push the module into the connector. Ensure that the DIMM snaps into the plastic locking levers on both ends.
5. Restore power to the system.

### 5.3 Increasing Microprocessor Speed

This section describes how to complete the following actions to increase microprocessor speed:

- Replace the Alpha 21164 microprocessor with an Alpha chip that has a higher speed rating.
- Reconfigure the clock divisor jumpers.

#### 5.3.1 Preparatory Information

**Caution:** Static-Sensitive Component – Due to the sensitive nature of electronic components to static electricity, anyone handling the microprocessor *must* wear a properly grounded antistatic wriststrap. Use of antistatic mats, ESD approved workstations, or exercising other good ESD practices is recommended.

An Alpha 21164 microprocessor with a higher speed rating is available from your local distributor. See Appendix A for information about supporting products.

When replacing the microprocessor chip, also replace the thermal conducting GRAFOIL pad. See Appendix A for information about the parts kit, which includes the heat sink, GRAFOIL pad, two hex nuts, heat-sink clips, 60-mm fan, fan guard, and four screws.

#### 5.3.2 Required Tools

The following tools are required when replacing the microprocessor chip:

A TS30 manual nut/torque driver (or equivalent) with the following attachments is required to affix the heat sink and fan to the microprocessor's IPGA package:

- 1/4-inch hex bit
- 7/16-inch socket with 1/4-inch hex drive
- #2 Phillips-head screwdriver bit

## Increasing Microprocessor Speed

### 5.3.3 Removing the 21164 Microprocessor

Remove the microprocessor currently in place at location U40 by performing the following steps:

1. Unplug the fan power/sensor cable from connector J18 (see Figure 2–1).
2. Remove the four 6-32 X 0.875-inch screws that secure the fan and fan guard to the heat sink.
3. Remove the fan and fan guard.
4. If the sink/chip/fan clip is used, remove it by unhooking its ends from around the ZIF socket retainers.
5. Using a 7/16-inch socket, remove the two nuts securing the heat sink to the microprocessor studs.
6. Remove the heat sink by gently lifting it off the microprocessor.
7. Remove and discard the GRAFOIL heat conduction pad.
8. Thoroughly clean the bottom surface of the heat sink before affixing it to the new microprocessor.
9. Lift the ZIF socket actuator handle to a full 90° angle.
10. Remove the microprocessor chip by lifting it straight out of the socket.

### 5.3.4 Installing the 21164 Microprocessor

Install the new microprocessor in location U40 by performing the following steps:

**Note:** Install the heat sink only after the microprocessor has been assembled to the ZIF socket.

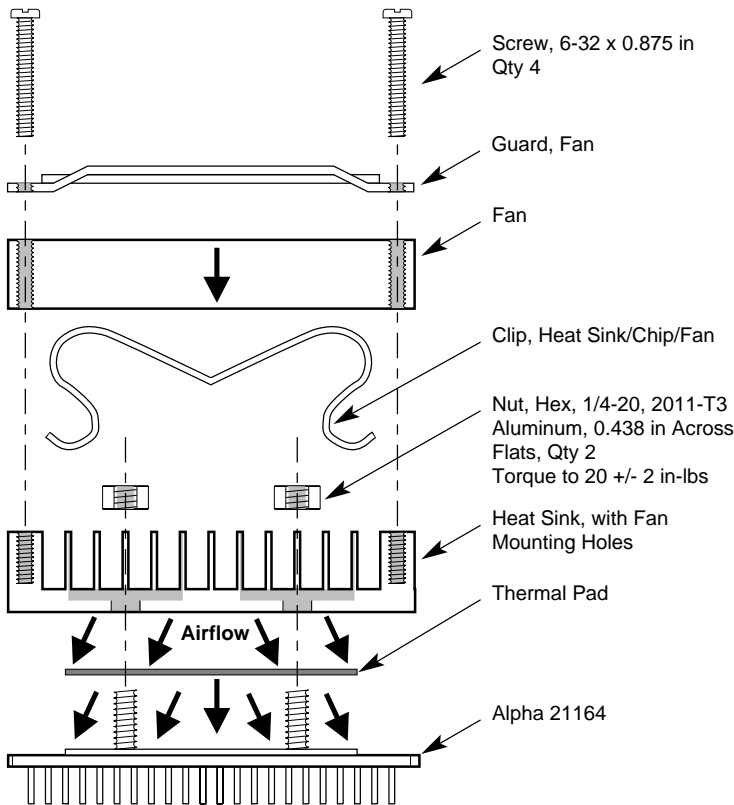
1. Observe antistatic precautions.
2. Lift the ZIF socket actuator handle to a full 90° angle.
3. Ensure that all the pins on the microprocessor package are straight.
4. The ZIF socket and microprocessor are keyed to allow for proper installation. Align the microprocessor, with its missing AD01 pin, with the corresponding plugged AD01 position on the ZIF socket. Gently lower into position.
5. Close the ZIF socket actuator handle to its locked position.

## Increasing Microprocessor Speed

6. Install the heat sink and heat-sink fan as directed in the following steps. A heat-sink/fan kit is available from the vendor listed in Appendix A. Refer to Figure 5–1 for heat-sink and fan assembly details.

Figure 5–1 shows the Fan/Heat-Sink Assembly on AlphaPC 164LX.

**Figure 5–1 Fan/Heat-Sink Assembly**



FM-06013.A14

- a. Put the GRAFOIL thermal pad in place. The GRAFOIL pad is used to improve the thermal conductivity between the chip package and the heat sink by replacing micro air pockets with a less insulative material. Perform the following steps to position the GRAFOIL pad:
  1. Perform a visual inspection of the package slug to ensure that it is free of contamination.
  2. Wearing clean gloves, pick up the GRAFOIL pad. *Do not* perform this with bare hands because skin oils can be transferred to the pad.

## Increasing Microprocessor Speed

3. Place the GRAFOIL pad on the gold-plated slug surface and align it with the threaded studs.
- b. Attach the microprocessor heat sink. The heat-sink material is clear anodized, hot-water-sealed, 6061-T6 aluminum. The nut material is 2011-T3 aluminum (this grade is critical). Perform the following steps to attach the heat sink:
1. Observe antistatic precautions.
  2. Align the heat-sink holes with the threaded studs on the ceramic package.
  3. Handle the heat sink by the edges and lower it onto the chip package, taking care not to damage the stud threads.
  4. Set a calibrated torque driver to 20 in-lbs,  $\pm 2$  in-lbs (2.3 Nm,  $\pm 0.2$  Nm). The torque driver should have a mounted 7/16-inch socket.
  5. Insert a nut into the 7/16-inch socket, place on one of the studs, and tighten to the specified torque. Repeat for the second nut.
  6. If the sink/chip/fan clip is used, properly install it by positioning it over the assembly and hooking its ends around the ZIF socket retainers.
- c. Attach the heat-sink fan assembly:
1. Place the fan assembly on top of the heat sink, aligning the fan mounting holes with the corresponding threaded heat-sink holes. Align the fan so that the fan power/sensor wires exit the fan closest to connector J18 (see Figure 2–1). Fan airflow must be directed into the heat sink (fan label facing down toward the heat sink).
  2. Place the fan guard on top of the fan. Orient the guard so that the corner mounting areas lay flush against the heat sink.
  3. Secure the fan and fan guard to the heat sink with four 6-32 X 0.875-inch screws.
  4. Plug the fan power/sensor cable into connector J18.

**Important:** When installing the microprocessor, you must change the frequency of its clock output by setting the system clock divisor jumpers, as described in Section 3.1.

# 6

---

## Interrupts and ISA Bus Addresses

This section lists the system and I/O interrupt assignments. It also lists the physical AlphaPC 164LX I/O space assignments.

### 6.1 Interrupts

Table 6–1 lists each AlphaPC 164LX ISA interrupt and its source.

**Table 6–1 ISA Interrupts**

<b>Interrupt Number</b>	<b>Interrupt Source</b>
IRQ0	Internal timer 1
IRQ1	Keyboard
IRQ2	Interrupt from controller 2
IRQ3	COM2
IRQ4	COM1
IRQ5	Available
IRQ6	Diskette (floppy)
IRQ7	Parallel port
IRQ8# <sup>1</sup>	Reserved
IRQ9	Available
IRQ10	Available
IRQ11	Available
IRQ12	Mouse

## ISA I/O Address Map

**Table 6–1 ISA Interrupts (Continued)**

Interrupt Number	Interrupt Source
IRQ13	Available
IRQ14	EIDE
IRQ15	EIDE

<sup>1</sup> The # symbol indicates an active low signal.

## 6.2 ISA I/O Address Map

Table 6–2 lists the AlphaPC 164LX ISA I/O space address mapping.

**Table 6–2 ISA I/O Address Map**

Range (hex)	Usage
000-00F	8237 DMA #1
020-021	8259 PIC #1
040-043	8253 timer
060-061	Ubus IRQ12 and NMI control
070	CMOS RAM address and NMI mask register
080-08F	DMA page registers
0A0-0A1	8259 PIC #2
0C0-0DF	8237 DMA #2
2F8-2FF	Serial port—COM2
370-377	Secondary diskette (floppy)
3BC-3BF	Parallel port—LPT1
3F0-3F7	Primary diskette (floppy)
3F8-3FF	Serial port—COM1
800	FLASH_ADR19 register
801	AlphaPC 164LX configuration register
804-806	PCI interrupt registers



### 6.2.1 Flash ROM Address Map

The address range for the flash ROM is FFF8.0000–FFFF.FFFF. Flash space of 1MB is obtained by double mapping this 512KB space. FLASH\_ADR19 register at I/O location  $800_{16}$  provides this function. Writing a 0 to this location enables the lower 512KB of flash. Writing a 1 to this location enables the upper 512KB of flash.



---

# Alpha SRM Console Firmware

The Alpha SRM Console firmware initializes the system and enables you to install and boot the Tru64 UNIX operating system. This firmware resides in the flash ROM on the AlphaPC 164LX motherboard.

## 7.1 Alpha SRM Console Firmware Conventions

The following conventions are used in this section:

<b>Convention</b>	<b>Description</b>
>>>	Alpha SRM Console prompt.
Backslash (\) at the end of a line	Continuation symbol to continue long commands on the next line.
_>	Continuation line prompt.
Maximum command length	255 characters.
Multiple contiguous spaces or tabs	Treated as a single space.
Command abbreviations	Allowed, if not ambiguous.
Command qualifiers or options	Prefix with a space and a dash (-).
Numbers	Hexadecimal, unless otherwise specified. (Registers, such as R0–R31, are shown in decimal notation.)

## Alpha SRM Console Firmware Conventions

The following table lists Alpha SRM Console special keys and their functions. These special keys, also referred to as shortcut keys, provide command recall, line editing, and basic input/output control flow.

Shortcut Key	Function
Enter	Terminate the command line input.
Backspace or Delete	Delete one character to the left of the cursor.
Ctrl/A	Toggles insert/overstrike mode. (Overstrike is the default.)
Ctrl/B Up arrow Down arrow	Recall previous commands. (The last 16 commands are stored.)
Ctrl/C	Terminate the foreground process.
Ctrl/D Left arrow	Move the cursor one position to the left.
Ctrl/E	Move the cursor to the end of the line.
Ctrl/F Right arrow	Move the cursor one position to the right.
Ctrl/H	Move the cursor to the beginning of the line.
Ctrl/O	Suppress or resume (toggle) console output.
Ctrl/Q	Resume the flow (XON) of data to the console.
Ctrl/R	Retype the current command line.
Ctrl/S	Stop the flow (XOFF) of data to the console.
Ctrl/U	Delete the entire line.

## 7.2 Basic Alpha SRM Console Command Descriptions

This section describes the following basic Alpha SRM Console commands that are necessary to boot the Tru64 UNIX operating system:

- arc
- boot
- deposit
- examine
- fwupdate
- set
- show

The Alpha SRM Console offers additional commands. For a complete list of Alpha SRM Console commands, enter **help** at the Alpha SRM Console prompt (>>>).

## Basic Alpha SRM Console Command Descriptions

### **arc**

Loads and runs the AlphaBIOS firmware update utility from a diskette.

### **Syntax**

**arc**

**nt**

### **Arguments**

None

### **Options**

None

### **Description**

None

### **Examples**

Either of the following commands load and run the AlphaBIOS firmware update utility from a diskette:

```
>>>arc
```

or

```
>>>nt
```

# Basic Alpha SRM Console Command Descriptions

## boot

Initializes the processor, loads a program image from the specified boot device, and transfers control to the loaded image.

### Syntax

```
boot [-file <filename>] [-flags  
<longword>[,<longword>]]  
[-protocols <enet_protocol>] [-halt]  
[<boot_device>]
```

### Arguments

<boot\_device>

A device path or list of devices from which the firmware will attempt to boot. Use the **set bootdef\_dev** command to set an environment variable that specifies a default boot device.

### Options

boot Command Option	Description
<b>-file</b> <filename>	Specifies the name of a file to load into the system. Use the <b>set boot_file</b> command to set the environment variable that specifies a default boot file.
<b>-flags</b> <longword> [,<longword>]	Specifies additional information for the operating system. For Tru64 UNIX systems, the following values may be used: i = Interactive boot s = Boot to single user a = Autoboot to multiuser Use the <b>set boot_osflags</b> command to set an environment variable that specifies a default boot flag value.
<b>-protocols</b> <enet_protocol>	Specifies the Ethernet protocols that will be used for a network boot. Values may be <b>mop</b> or <b>bootp</b> .
<b>-halt</b>	Forces the bootstrap operation to halt and invoke the console program after the image is loaded and the page tables and other data structures are set up.

## Basic Alpha SRM Console Command Descriptions

### Description

The **boot** command initializes the processor, loads a program image from the specified boot device, and transfers control to that image. If you do not specify a boot device in the command line, the default boot device is used. The default boot device is determined by the value of the `bootdef_dev` environment variable.

If you specify a list of devices, a bootstrap is attempted from each device in the order in which the device is listed. Then control passes to the first successfully booted image. In a list, always enter network devices *last* because network bootstraps terminate only if a fatal error occurs or if an image is successfully loaded.

The **-flags** option can pass additional information to the operating system about the boot that you are requesting. On an OpenVMS system, the **-flags** option specifies the system root number and boot flags. If you do not specify a boot flag qualifier, the default boot flag's value specified by the `boot_osflags` environment variable is used.

The **-protocols** option allows selection of either the DECnet MOP or the TCP/IP BOOTP network protocols. The keywords **mop** and **bootp** are valid arguments for this option. It is possible to set the default protocol for a port by setting the environment variable `ewa0_protocols` or `era0_protocols` to the appropriate protocol.

Explicitly stating the boot flags or the boot device overrides the current default value for the current boot request, but does not change the corresponding environment variable.

See the Environment Variables for Alpha SRM Console Commands section in this chapter for more information about environment variables.



# Basic Alpha SRM Console Command Descriptions

## Examples

---

<b>boot Command Example</b>	<b>Description</b>
<code>&gt;&gt;&gt;boot</code>	Boots the system from the default boot device.
<code>&gt;&gt;&gt;boot ewa0</code>	Boots the system from Ethernet port ewa0.
<code>&gt;&gt;&gt;boot -file dec2.sys ewa0</code>	Boots the file named dec2.sys from Ethernet port ewa0.
<code>&gt;&gt;&gt;boot -protocol bootp ewa0</code>	Boots the system using the TCP/IP BOOTP protocol from Ethernet port ewa0.
<code>&gt;&gt;&gt;boot -flags 0,1</code>	Boots the system from the default boot device using flag setting 0,1.
<code>&gt;&gt;&gt;boot -halt dka0</code>	Loads the bootstrap image from disk dka0, halts the bootstrap operation, and invokes the console program. Subsequently, you can enter <b>continue</b> to transfer control to the operating system.

---

# Basic Alpha SRM Console Command Descriptions

## deposit

Writes data to the specified address.

### Syntax

```
deposit [-{b,w,l,q,o,h}] [{physical, virtual, gpr, fpr, ipr}] [-n <count>] [-s <step>] [<device>:]<address> <data>
```

### Arguments

**<device>**:

The optional device name (or address space) selects the device to access. The following platform-independent devices are supported:

- **pmem**  
Physical memory.
- **vmem**  
Virtual memory. All access and protection checking occur. If the access is not allowed to a program running with the current processor status (PS), the console issues an error message. If memory mapping is not enabled, virtual addresses are equal to physical addresses.

**<address>**

An address that specifies the offset within a device into which data is deposited. The address may be any legal symbolic address.

Valid symbolic addresses are shown in the following table.

---

Symbolic Address	Description
<b>gpr-name</b>	Represents general-purpose register.
<b>ipr-name</b>	Represents internal processor register.
<b>PC</b>	Program counter.
<b>+</b>	The location immediately following the last location referenced by <b>examine</b> or <b>deposit</b> .

# Basic Alpha SRM Console Command Descriptions

---

Symbolic Address	Description
-	The location immediately preceding the last location referenced by <b>examine</b> or <b>deposit</b> .
*	The location last referenced by <b>examine</b> or <b>deposit</b> .
@	The location addressed by the last location referenced by <b>examine</b> or <b>deposit</b> .

---

<data>

The data to be deposited.

## Options

---

deposit Command Option	Description
-b	Specifies data type is byte.
-w	Specifies data type is word.
-l	Specifies data type is longword.
-q	Specifies data type is quadword.
-o	Specifies data type is octaword.
-h	Specifies data type is hexword.
-physical	References physical address space.
-virtual	References virtual address space.
-gpr	References general-purpose register address space.
-fpr	References floating-point register address space.
-ipr	References internal processor register address space.
-n <count>	Specifies the number of consecutive locations to examine.
-s <step>	Specifies the address increment as a hexadecimal value. This option allows you to override the increment that is normally derived from the data size.

---

# Basic Alpha SRM Console Command Descriptions

## Description

The **deposit** command writes data to the specified address, such as a memory location, register, device, or file. The defaults for address space, data size, and address are the last specified values. After initialization, the default for address space is physical memory; for data size, the default is a quadword; and for address, the default is zero.

An address or device can be specified by concatenating the device name with the address. For example, use **pmem:0** and specify the size of the address space to be written. If a conflicting device, address, or data size is specified, the console ignores the command and issues an error response.

## Examples

<b>deposit Command Example</b>	<b>Description</b>
<code>&gt;&gt;&gt;d -n 1ff pmem:0 0</code>	Clears the first 512 bytes of physical memory.
<code>&gt;&gt;&gt;d -l -n 3 pmem:1234 5</code>	Writes the value 5 into four longwords, starting at physical memory address 1234.
<code>&gt;&gt;&gt;d -n 8 r0 ffffffff</code>	Loads GPRs R0 through R8 with -1.
<code>&gt;&gt;&gt;d -l -n 10 -s 200 pmem:0 8</code>	Writes the value 8 in the first longword of the first 17 pages in physical memory.

# Basic Alpha SRM Console Command Descriptions

## examine

Displays the contents of the specified address.

### Syntax

```
examine [-{b,w,l,q,o,h,d}] [-{physical, virtual,  
gpr, fpr, ipr}] [-n <count>] [-s <step>]  
[<device>:]<address>
```

### Arguments

<device>:

The optional device name (or address space) selects the device to access.

<address>

The address specifies the first location to examine within the current device. The address can be any legal address specified.

### Options

<b>examine Command Option</b>	<b>Description</b>
<b>-b</b>	Specifies data type is byte.
<b>-w</b>	Specifies data type is word.
<b>-l</b>	Specifies data type is longword.
<b>-q</b>	Specifies data type is quadword.
<b>-o</b>	Specifies data type is octaword.
<b>-h</b>	Specifies data type is hexword.
<b>-d</b>	Specifies the data displayed is the decoded macro instruction. The Alpha instruction decode (-d) does not recognize machine-specific PALcode instructions.
<b>-physical</b>	References physical address space.
<b>-virtual</b>	References virtual address space.
<b>-gpr</b>	References general-purpose register address space.
<b>-fpr</b>	References floating-point register address space.

## Basic Alpha SRM Console Command Descriptions

<b>examine Command Option</b>	<b>Description</b>
<b>-ipr</b>	References internal processor register address space.
<b>-n &lt;count&gt;</b>	Specifies the number of consecutive locations to examine.
<b>-s &lt;step&gt;</b>	Specifies the address increment as a hexadecimal value. This option allows you to override the increment that is normally derived from the data size.

### Description

The **examine** command displays the contents of the specified address, such as a memory location, register, device, or file. The defaults for address space, data size, and address are the last specified values. After initialization, the default for address space is physical memory; for data size, the default is a quadword; and for address, the default is zero.

An address or device can be specified by concatenating the device name with the address. For example, use **pmem:0** and specify the size of the address space to be displayed. If a conflicting device, address, or data size is specified, the console ignores the command and issues an error response.

The display line consists of the device name, the hexadecimal address (or offset within the device), and the examined data (also in hexadecimal).

The **examine** command supports the same options as the **deposit** command. Additionally, the **examine** command supports instruction decoding with the **-d** option, which disassembles instructions beginning at the current address.

# Basic Alpha SRM Console Command Descriptions

## Examples

---

<b>examine Command Example</b>	<b>Display</b>	<b>Description</b>
<code>&gt;&gt;&gt;e r0</code>	<code>gpr: 0 (R0) 0000000000000002</code>	Examines the contents of R0, using a symbolic address.
<code>&gt;&gt;&gt;e -g 0</code>	<code>gpr: 0 (R0) 0000000000000002</code>	Examines the contents of R0, using address space.
<code>&gt;&gt;&gt;e grp:0</code>	<code>gpr: 0 (R0) 0000000000000002</code>	Examines the contents of R0, using a device name.
<code>&gt;&gt;&gt;examine -n 5 r7</code>	<code>gpr: 38 (R7) 0000000000000000 gpr: 40 (R8) 0000000000000000 gpr: 48 (R9) 0000000000000000 gpr: 50 (R10) 000000007FFBF800 gpr: 58 (R11) 000000007FF781A2 gpr: 60 (R12) 0000000000000000</code>	Examines the contents of R7 and the next five registers.
<code>&gt;&gt;&gt;examine ipr:11</code>	<code>ipr 11 (KSP) FFFFFFFF8228DFD0</code>	Examines the contents of internal processor register 11.

---

## Basic Alpha SRM Console Command Descriptions

### **fwupdate**

Loads and runs the AlphaBIOS firmware update utility from a diskette.

#### **Syntax**

**fwupdate**

#### **Arguments**

None

#### **Options**

None

#### **Description**

The **fwupdate** command script is used to load and run the AlphaBIOS firmware update utility from a diskette. The file `fwupdate.exe` is extracted from a diskette with a FAT file structure. This executable is then loaded to physical address 900000 and is executed in PALmode.

#### **Examples**

The following **fwupdate** command script loads and runs the AlphaBIOS firmware update utility from a diskette:

```
>>>fwupdate
```



# Basic Alpha SRM Console Command Descriptions

## set

Sets or modifies the value of an environment variable.

### Syntax

```
set <envar> <value> [-default] [-integer] [-string]
```

### Arguments

<envar>

The environment variable to be assigned a new value.

<value>

The value that is assigned to the environment variable. It can be either a numeric value or an ASCII string.

### Options

set Command Option	Description
-default	Restores an environment variable to its default value.
-integer	Creates an environment variable as an integer.
-string	Creates an environment variable as a string.

### Description

The **set** command is used to set or modify the value of an environment variable. Environment variables are used to pass configuration information between the console and the operating system. See Section 7.3 for more information about environment variables.

## Basic Alpha SRM Console Command Descriptions

### Examples

<b>set Command Example</b>	<b>Description</b>
<code>&gt;&gt;&gt;set bootdef_dev ewa0</code>	Modifies the default boot device to ewa0.
<code>&gt;&gt;&gt;set auto_action boot</code>	Attempts to boot the operating system following an error, halt, or power-up.
<code>&gt;&gt;&gt;set boot_osflags 0,1</code>	Modifies the default boot flags to 0,1.
<code>&gt;&gt;&gt;set foobar 5</code>	Creates an environment variable called foobar and gives it a value of 5.

# Basic Alpha SRM Console Command Descriptions

## show

Displays the current value of the specified environment variable or information about the system.

### Syntax

```
show [{config, device [device_name], iobq, hwrpb,  
map, memory, pal, version, <envar>...}]
```

### Arguments

show Command Argument	Description
<b>config</b>	Displays the current memory configuration, PCI logical slots, and ISA logical slots.
<b>device</b> [ <b>device name</b> ]	Displays the devices and controllers in the system. Specifying a device name returns information on that device only.
<b>iobq</b>	Displays the input/output counter blocks.
<b>hwrpb</b>	Displays the hardware restart parameter block.
<b>map</b>	Displays the system virtual memory map.
<b>memory</b>	Displays the memory module configuration.
<b>pal</b>	Displays the version of Tru64 UNIX PALcode.
<b>version</b>	Displays the version of the console.
<b>&lt;envar&gt;</b>	Displays the current value of a specified environment variable.

### Options

None

### Description

The **show** command displays information about the system and the current value of a specified environment variable. See Section 7.3 for more information about environment variables.

# Environment Variables for Alpha SRM Console Commands

## Examples

show Command Example	Description
<pre>&gt;&gt;&gt;show device dka0.0.0.6.0      DKA0   RZ26L  441A dka400.4.0.6.0   DKA400 RRD43  3213 dva0.0.0.0.1     DVA0 ewa0.0.0.12.0    EWA0   08-00-2B-E2-1C-25 pka0.7.0.6.0     PKA0   SCSI Bus ID 7</pre>	Lists device information, such as system designation, drive model, or Ethernet address.
<pre>&gt;&gt;&gt;show memory 48 Meg of System Memory</pre>	Lists system random-access memory (RAM) size.
<pre>&gt;&gt;&gt;show * (refer to Section 7.3)</pre>	Lists all environment variables and their settings.
<pre>&gt;&gt;&gt;show boot* (refer to Section 7.3)</pre>	Lists all environment variables, beginning with boot.

## 7.3 Environment Variables for Alpha SRM Console Commands

This section describes environment variables that are used to define the system operational state and to pass information between the firmware and the operating system.

### 7.3.1 Environment Variable Descriptions

Environment variables are classified as either Alpha SRM Console architecture-required or system-defined.

# Environment Variables for Alpha SRM Console Commands

## 7.3.1.1 Architecture-Required Environment Variables

The following table shows common Alpha SRM Console architecture-required environment variables and their descriptions. For a complete list, enter **show \*** at the Alpha SRM Console prompt.

Architecture-Required Environment Variable	Description
<code>auto_action</code>	When used with the <b>set</b> or <b>show</b> command, this variable modifies or displays the console action that follows an error, halt, or power-up. The action can be halt, boot, or restart. The default is halt.
<code>boot_file</code>	When used with the <b>set</b> or <b>show</b> command, this variable modifies or displays the file name to be used when a bootstrap requires a file name. The default is null.
<code>boot_osflags</code>	When used with the <b>set</b> or <b>show</b> command, this variable modifies or displays the additional parameters to be passed to system software. The default is 0.
<code>bootdef_dev</code>	When used with the <b>set</b> or <b>show</b> command, this variable modifies or displays the default device or device list from which the system will attempt to boot. If the system software is preloaded, the variable is preset to point to the device containing the preloaded software. The default is null.

## Environment Variables for Alpha SRM Console Commands

### 7.3.1.2 System-Defined Environment Variables

The following table shows common Alpha SRM Console system-defined environment variables and their descriptions. For a complete list, enter **show \*** at the Alpha SRM Console prompt.

System-Defined Environment Variable	Description
<code>console</code>	When used with the <b>set</b> command, this variable modifies the console output to either the serial port or the graphics controller.
<code>control_scsi_term</code>	This variable is unused in the motherboard system.
<code>ewa0_mode</code>	This variable determines if the AUI (ThinWire) or the twisted-pair Ethernet ports will be enabled. AUI is the default. (Autosensing is not supported.)
<code>os_type</code>	When used with the <b>set</b> or <b>show</b> command, this variable modifies or displays the specified firmware that will be loaded on the next power cycle. Specify the value <b>osf</b> or <b>UNIX</b> to select the Alpha SRM Console.
<code>pci_parity</code>	This variable controls PCI parity checking. The possible values are: <b>on</b> = Parity checking is enabled. <b>off</b> = Parity checking is disabled; this is the default. <b>sniff</b> = Parity checking is enabled or disabled depending on the PCI device.
<code>oem_string</code>	When used with the <b>set</b> or <b>show</b> command, this variable modifies or displays a text string that identifies the product name in the Alpha SRM Console banner.

# Environment Variables for Alpha SRM Console Commands

---

System-Defined Environment Variable	Description
<code>language n</code>	<p>The language environment variable assigns language <code>n</code> to the system (where <code>n</code> is the option number of a language listed in the menu that follows). Use the following procedure to select the language:</p> <ol style="list-style-type: none"><li>At the Alpha SRM Console prompt, enter the following commands: <pre>&gt;&gt;&gt;set language 0 &gt;&gt;&gt;init</pre></li></ol> <p>The following menu and prompt are displayed:</p> <pre>n Language                n Language =====   0 none (display menu)    40 Français (Suisse Romande)  30 Dansk                  42 Italiano  32 Deutsch                44 Nederlands  34 Deutsch (Schweiz)     46 Norsk  36 English (American)    48 Portugues  38 English (British/Irish) 4A Suomi  3A Español                4C Svenska  3C Français              4E Vlaams  3E Français (Canadian)</pre> <p>( 1 . . 16 ) :</p> <ol style="list-style-type: none"><li>Enter the number that corresponds to the language that you want to use. The following example shows how to assign the English (American) language to the system: <pre>( 1 . . 16 ) : 36</pre></li><li>When you receive a message to reset the system, power-cycle the system.</li></ol>

---

# Using the AlphaBIOS Firmware Update Utility to Update the Flash ROM

## 7.4 Using the AlphaBIOS Firmware Update Utility to Update the Flash ROM

Use the AlphaBIOS firmware update utility to update the firmware in a flash ROM.

### 7.4.1 AlphaBIOS Conventions

AlphaBIOS uses universally accepted keys and key combinations for navigating the interface and selecting items. If you are familiar with MS-DOS or Microsoft Windows keyboard conventions, navigating AlphaBIOS is simple. Use the keys and key combinations shown in Table 7–1 when navigating and selecting items in AlphaBIOS.

**Table 7–1 AlphaBIOS Keys**

Key or Key Combination	Description
Tab	Move highlight forward between fields of a dialog.
Shift + Tab	Move highlight backwards between fields of a dialog.
↓ or ↑	Move highlight within a menu, or cycle through available field values in a dialog window.
Alt + ↓	Drop down a menu of choices from a drop-down listbox. A drop-down listbox can be recognized by the symbol ↓.
Home	Move to the beginning of a text-entry field.
End	Move to the end of a text-entry field.
← or →	Move to the left or right in a text-entry field.
Esc	Discard changes and back up to previous screen.

Two levels of keyboard help are available:

- Press **F1** once to display explanations of the keystrokes available for the currently displayed part of AlphaBIOS.
- Press **F1** twice to display explanations of the keystrokes available for navigating throughout AlphaBIOS.



# Using the AlphaBIOS Firmware Update Utility to Update the Flash ROM

## 7.4.2 Starting the AlphaBIOS

To start the AlphaBIOS, follow this procedure:

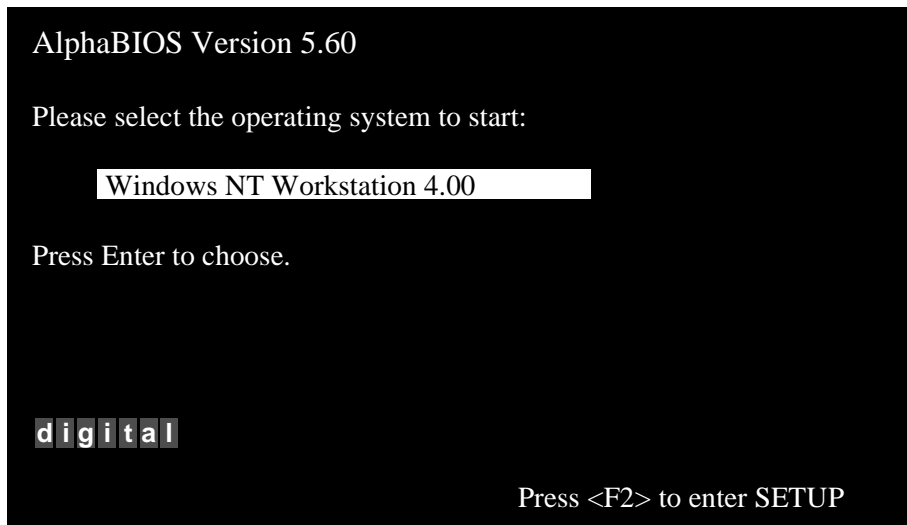
1. Insert the AlphaBIOS diskette into diskette drive A.
2. At the Alpha SRM Console prompt, enter the following command:

```
>>>fwupdate
```

**Note:** Because the firmware update utility reinitializes some system components, it may appear as if your system is restarting.

Figure 7–1 shows an example of the AlphaBIOS Boot screen with the “Press <F2> to enter SETUP” message at the bottom.

**Figure 7–1 AlphaBIOS Boot Screen**



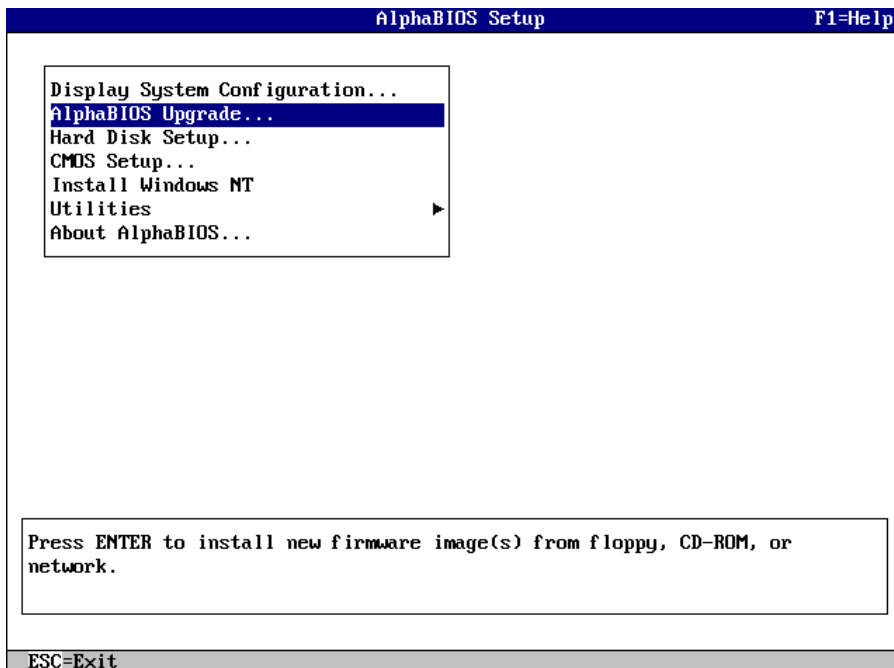
3. Press **F2** to start the AlphaBIOS setup program.

# Using the AlphaBIOS Firmware Update Utility to Update the Flash ROM

## 7.4.3 Installing Alpha SRM Console Using AlphaBIOS Setup Program

Figure 7–2 shows an example of the AlphaBIOS Setup screen. Select the **AlphaBIOS Upgrade...** option by using the arrow or Tab keys.

Figure 7–2 AlphaBIOS Setup Screen

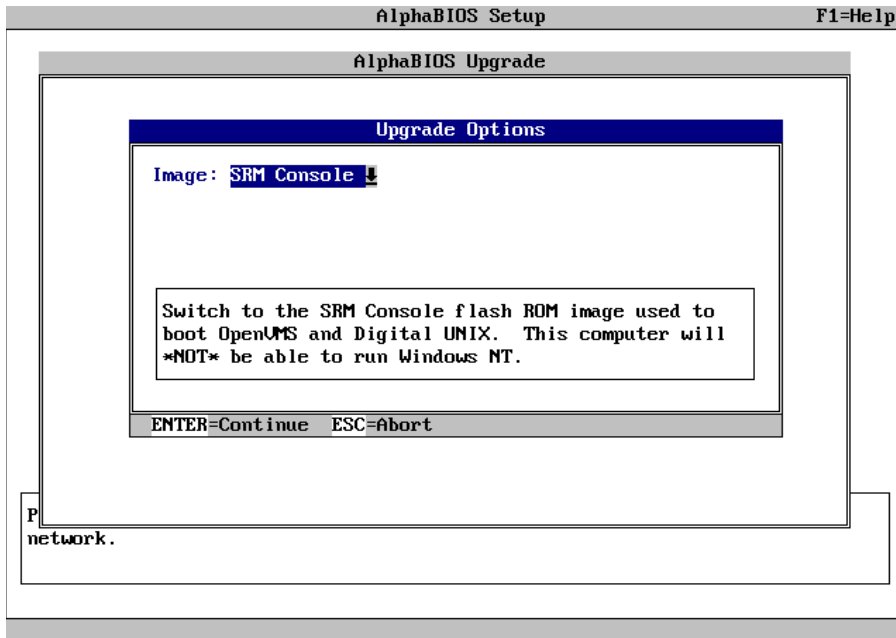


Press **Enter** to begin installing the SRM Console firmware image.

# Using the AlphaBIOS Firmware Update Utility to Update the Flash ROM

Figure 7–3 shows an example of the AlphaBIOS Upgrade Options screen. If more than one image is found, the new image's name is displayed. If the name of the new image is not SRM Console, use the down arrow key to cycle through the available field values until SRM Console is displayed.

**Figure 7–3 AlphaBIOS Upgrade Options Screen**

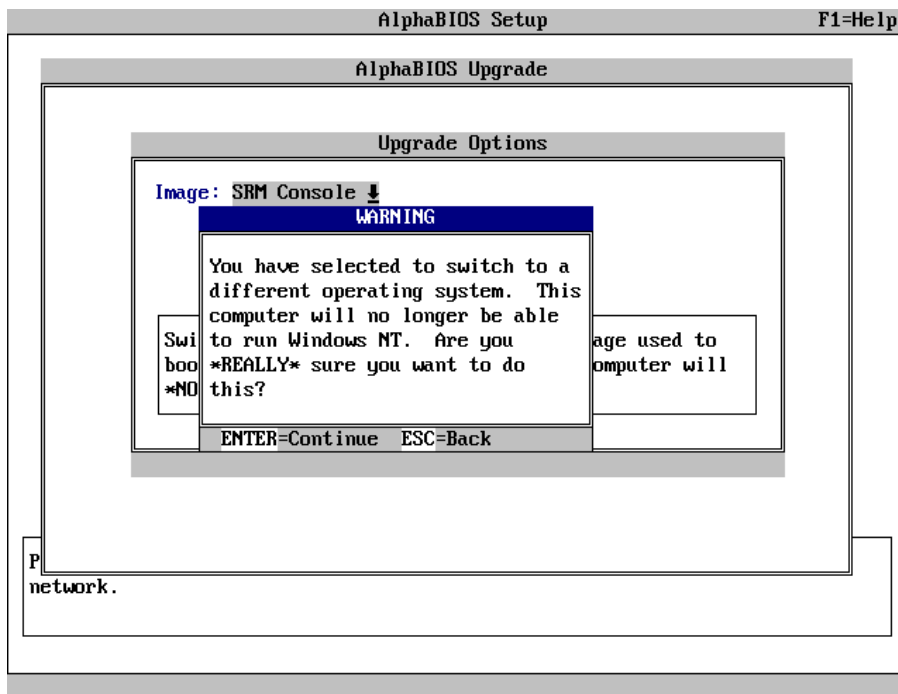


Press **Enter** to continue the installation.

# Using the AlphaBIOS Firmware Update Utility to Update the Flash ROM

Figure 7–4 shows an example of the AlphaBIOS screen that warns you that you have selected to switch the operating system.

**Figure 7–4 AlphaBIOS Warning Screen**

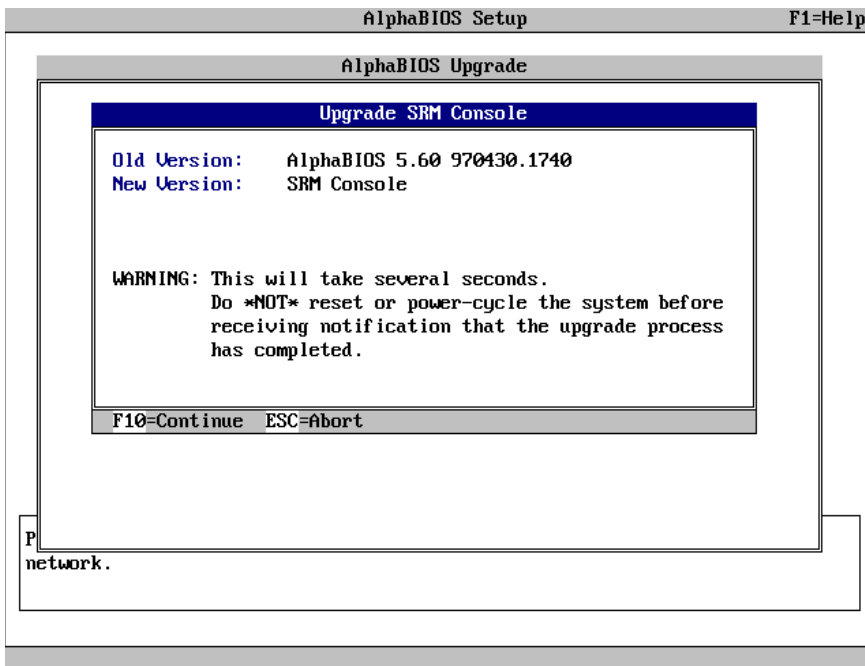


Press **Enter** to continue the installation.

# Using the AlphaBIOS Firmware Update Utility to Update the Flash ROM

A screen similar to Figure 7–5 is displayed. The version numbers shown on your screen may be different than those shown in Figure 7–5.

**Figure 7–5 AlphaBIOS Upgrade SRM Console Screen**

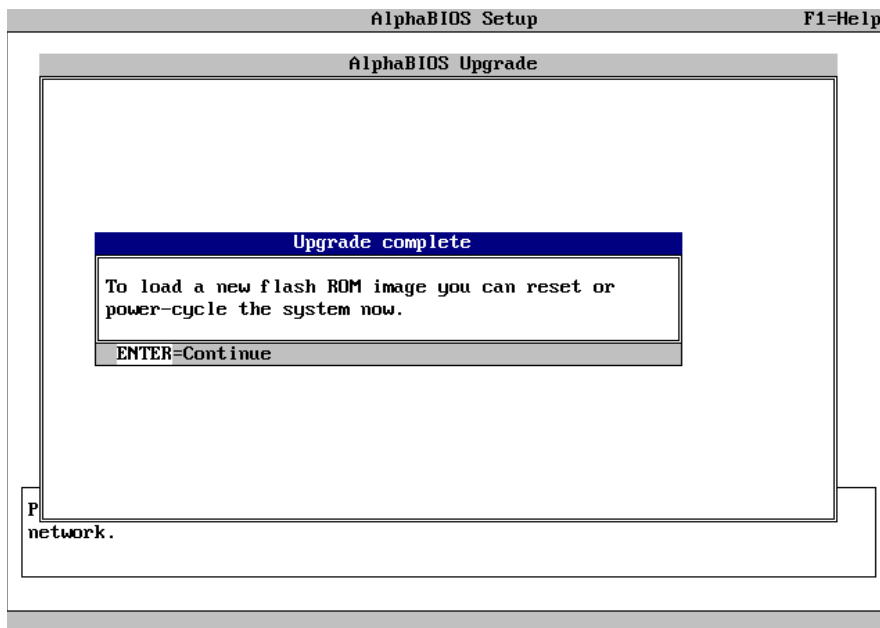


Press **F10** to continue the installation.

# Using the AlphaBIOS Firmware Update Utility to Update the Flash ROM

Figure 7–6 shows an example of the AlphaBIOS Upgrade Complete screen.

**Figure 7–6 AlphaBIOS Upgrade Complete Screen**



To load the SRM Console, power-cycle the system.

## 7.5 Installing the Tru64 UNIX Operating System

**Note:** The DIGITAL UNIX product name has been changed to Tru64 UNIX. Some references to older documentation titles use the DIGITAL UNIX product name.

**Note:** Tru64 UNIX Version 4.0D is the minimum revision required for the AlphaPC 164LX motherboard system.

This section supplements the *DIGITAL UNIX Installation Guide* for installing the Tru64 UNIX operating system on an AlphaPC 164LX motherboard system.

**Note:** If you already have a previous version of Tru64 UNIX installed, see the *DIGITAL UNIX Installation Guide* for information on how to upgrade to a new version of the operating system. Be sure to review the preinstallation tasks, which are covered in the *DIGITAL UNIX Installation Guide*.

### 7.5.1 Requirements

You need the following hardware and software to install the Tru64 UNIX operating system on an AlphaPC 164LX motherboard system:

- A minimum of 32MB of main memory; 64MB is recommended
- A 1GB (or larger) SCSI or EIDE hard disk capable of storing the supported software subsets
- Supported load devices
  - SCSI CD-ROM drive capable of reliably reading in 512-byte block mode  
or
  - EIDE (ATAPI) CD-ROM drive  
or
  - A network interface
- A console terminal with ASCII capability or a supported graphics display console
- Tru64 UNIX Version 4.0D or higher Operating System Volume 1 compact disc
- Alpha SRM Console Version 4.9 or higher





This chapter contains information about troubleshooting hardware and software during AlphaPC 164LX startup.

## 8.1 Hardware Startup

Use the following troubleshooting steps if video is not working on your system. If you still have no video after reviewing these steps, please call your system vendor.

### **Troubleshooting Steps: No Video**

1. Check the connection to the ac outlet.
2. Check the voltage setting on the power supply (115 V ac in the U.S.).
3. Check that the frequency/jumper selection matches the speed of the Alpha chip.
4. Check that the oscillator is installed into the motherboard correctly.
5. Check that the CPU fan is connected and spinning.
6. Ensure that the flash ROM update procedure was performed correctly. If you have a terminal attached to COM1, check the output for error messages after verifying that the flash ROM update procedure was performed correctly.
7. Reseat the video card and ensure that it is connected to the monitor.
8. Reseat the DIMMs.
9. Replace the DIMMs.

## Beep Codes

### 8.2 Beep Codes

The beep codes provide error information about the AlphaPC 164LX system. Table 8–1 lists and describes the beep codes.

**Table 8–1 Beep Codes**

Beep Code	Description
1–2–3 <sup>1</sup>	This sequence represents the fail-safe booter startup.
4	No valid header found in ROM; loading entire ROM.
5	No memory found.
6	Checksum error detected when image was read back from memory.

<sup>1</sup> One beep and a pause, followed by two beeps and a pause, followed by three beeps.

### 8.3 Post Codes

The post codes indicate the progress of the SROM and SRM Console firmware. Table 8–2 lists and describes the post codes.

**Table 8–2 Post Codes**

Source	Post Code (hex)	Description
SROM	00	Firmware initialization is complete
	01	CPU speed detected
	02	CPU speed converted
	03	Configuration jumpers read
	04	Bcache configuration value computed
	05	Bcache control value computed
	06	Bcache turned off
	0C	Memory sized and memory bank 0 written
	0F	Bcache turned on
	13	All of memory rewritten (good data parity written)
	14	Memory errors cleared; start reading system ROM
	15	Loading ROM without SROM decompression
	16	Loading ROM using SROM decompression
	17	System ROM loaded to memory
	18	Icache flush code written to memory
	19	CPU errors cleared; jump to system code

Table 8–2 Post Codes (Continued)

Source	Post Code (hex)	Description
	20	ISA bus reset
	3F	Fatal error. Second code identifies source of error: 05 = No memory found 06 = Checksum error detected when image was read back from memory
SRM Console	FF	Starting console
	FE	Idle PCB initialization
	FD	Semaphore initialization
	FC	Heap initialization
	FB	Heap initialization
	FA	Heap initialization
	F9	Driver structure initialization
	F8	Idle process PID initialization
	F7	File system initialization
	F6	Timer data structures initialization
	F5	Lowering IPL
	F4	Entering idle loop
	F3	Creating task to deallocate dead PCBs
	F2	Creating polling task
	F1	Creating timer task
	F0	Creating power-up task
	EF	Configuring memory
	EE	Phase 1 driver startup
	ED	Configuring the PCI/ISA bus
	EC	Phase 3 driver startup
	EB	Switching stdin/out/err to console terminal device
	EA	Phase 4 driver startup
	E9	Building per CPU slot in the HWRPB
	E8	SCSI class driver initialization
	E7	Phase 5 driver startup
Fail-safe booter	BF	Fail-safe booter startup

### 8.4 Fail-Safe Booter

The fail-safe booter provides an emergency recovery mechanism when the primary firmware image contained in flash memory has been corrupted. When flash memory has been corrupted, and no image can be loaded safely from the flash, you can run the fail-safe booter and boot another image from a diskette that is capable of reprogramming the flash.

#### Starting the Fail-Safe Booter

You can start the fail-safe booter in one of two ways:

- If the primary firmware image is unavailable when the system is powered on or reset, the fail-safe booter runs automatically. When the fail-safe booter runs, the system emits a series of beeps through the speaker as beep code 1-2-3; that is, one beep and a pause, followed by two beeps and a pause, followed by three beeps. After the diskette activity light flashes, insert the *AlphaPC 164LX SRM Console Firmware* diskette. The fail-safe booter will load and run the AlphaBIOS firmware update utility from this diskette. Proceed to Section 7.4 and follow the procedure for updating your flash ROM.
- You can also start the fail-safe booter manually as follows:
  1. Power off your system.
  2. Add jumper CF7 as described in Section 3.3.
  3. Power on your system.
  4. Insert the *AlphaPC 164LX SRM Console Firmware* diskette into the diskette drive.
  5. Proceed to Section 7.4 and follow the procedures.

---

## Battery Recycle/Disposal Information

### NOTICE

Recycle or dispose of batteries promptly in accordance with your organization's environmental policies. If this is a LITHIUM battery, the following additional precautions may apply:

- Replace batteries correctly to prevent possible explosion.
- Replace batteries with the same or equivalent type.
- Prior to disposal or recycling, protect all batteries against accidental short circuiting by affixing nonconductive tape across battery terminals or conductive surfaces.
- Keep batteries away from small children.



---

# A

## Support

### A.1 Customer Support

The Alpha OEM website provides the following information for customer support.

URL	Description
<a href="http://www.digital.com/alphaoem">http://www.digital.com/alphaoem</a>	Contains the following links: <ul style="list-style-type: none"><li>• <b>Developers' Area:</b> Development tools, code examples, driver developers' information, and technical white papers</li><li>• <b>Motherboard Products:</b> Motherboard details and performance information</li><li>• <b>Microprocessor products:</b> Microprocessor details and performance information</li><li>• <b>News:</b> Press releases</li><li>• <b>Technical Information:</b> Motherboard firmware and drivers, hardware compatibility lists, and product documentation library</li><li>• <b>Customer Support:</b> Feedback form</li></ul>

---

## Supporting Products

### A.2 Supporting Products

This section lists sources for components and accessories that are not included with the AlphaPC 164LX.

#### A.2.1 Memory

Dual inline memory modules (DIMMs) are available from a variety of vendors. For a list of qualified vendors, visit the Alpha OEM World Wide Web Internet site at URL:

<http://www.digital.com/alphaoem>

Click on **Technical Information**.

Then click on **Alpha OEM Hardware Compatibility List**.

#### A.2.2 Thermal Products

Components included in this heat-sink and fan solution are heat sink, GRAFOIL pad, two hex nuts, heat-sink clips, 60-mm fan, and four screws. These are available from:

**United Machine and Tool Design Company, Inc.**

18 River Road  
P.O. Box 168  
Fremont, NH 03044  
Phone: 603-642-5040  
Fax: 603-642-5819  
PN 70-32810-02

#### A.2.3 Power Supply

An ATX form-factor power supply, suitable for use with the AlphaPC 164LX (+3.3 V, +5 V, -5 V, +12 V, -12 V), is available from:

**Quantum Power Labs, Inc.**

1410 Gail Borden Place C-4  
El Paso, TX 79935  
Phone: 915-599-2688  
Fax: 915-599-2699  
PN 11-006A (300 W)



## Antec, Inc.

2859 Bayview Drive  
Fremont, CA 94538  
Phone: 510-770-1200, ext. 313  
Contact: Han Liu  
PN PP-253X

## A.2.4 Enclosure

An enclosure, suitable for housing the AlphaPC 164LX and its power supply, is available from:

### Delta Axxion Technology

1550 Northwestern Drive  
El Paso, TX 79912  
Phone: 915-225-8888  
PN DL17

## A.3 Alpha Documentation

The following table lists some of the available Alpha documentation. You can download Alpha documentation from the Alpha OEM World Wide Web Internet site:

<http://www.digital.com/alphaoem>

Click on **Technical Information**.

Then click on **Documentation Library**.

Title	Order Number
Alpha Architecture Reference Manual <sup>1</sup>	EY-W938E-DP
Alpha Architecture Handbook	EC-QD2KB-TE
Alpha 21164 Microprocessor Hardware Reference Manual	EC-QP99C-TE
Alpha 21164 Microprocessor Data Sheet	EC-QP98C-TE

<sup>1</sup> Not available on website. To purchase the *Alpha Architecture Reference Manual*, contact your sales office or call Butterworth-Heinemann (DIGITAL Press) at 1-800-366-2665.

## Third-Party Documentation

### A.4 Third-Party Documentation

You can order the following third-party documentation directly from the vendor.

<b>Title</b>	<b>Vendor</b>
PCI Local Bus Specification, Revision 2.1	PCI Special Interest Group
PCI Multimedia Design Guide, Revision 1.0	U.S. 1-800-433-5177
PCI System Design Guide	International 1-503-797-4207
PCI-to-PCI Bridge Architecture Specification, Revision 1.0	Fax 1-503-234-6762
PCI BIOS Specification, Revision 2.1	
82420/82430 PCIset ISA and EISA Bridges (includes 82378IB/ZB SIO) (PN 290483)	Intel Corporation Literature Sales P.O. Box 7641 Mt. Prospect, IL 60056 Phone: 1-800-628-8686 FaxBACK Service: 1-800-628-2283 BBS: 1-916-356-3600
Super I/O Combination Controller (FDC37C935) Data Sheet	Standard Microsystems Corporation 80 Arkay Drive Hauppauge, NY 11788 Phone: 1-516-435-6000 Fax: 1-516-231-6004