AlphaPC 164SX Motherboard DIGITAL UNIX

User's Manual

Order Number: EC–R8P7A–TE

Revision/Update Information: This is a new document.

Contents

5   Memory and Microprocessor Configuration

6   Interrupts and ISA Bus Addresses

7   Alpha SRM Console Firmware

8   Troubleshooting

9 Battery Recycle/Disposal Information

A Supporting Products

B Associated Documentation

Figures

Tables

1

About This Manual

This manual describes the AlphaPC 164SX motherboard, a board for computing systems based on the DIGITAL Semiconductor Alpha 21164PC microprocessor and the companion DIGITAL Semiconductor 21174 core logic chip. It describes the motherboard's features and how to set its configuration switches. This manual helps users to install and populate the AlphaPC 164SX motherboard with memory boards 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 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:

This term...	Refers to...
Alpha SRM console	The Alpha SRM Console firmware
DIGITAL UNIX installation guide	The *DIGITAL UNIX Installation Guide*
DIGITAL UNIX	The DIGITAL UNIX operating system

2

Features

Table 2–1 provides an overview of the AlphaPC 164SX motherboard's features.

Table 2–1 AlphaPC 164SX Features

Feature	Description
Microprocessor	DIGITAL Semiconductor Alpha 21164PC microprocessor (64-bit RISC)
Core logic chip	DIGITAL Semiconductor 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 512MB memory array -- Two banks of 128-bit memory; 168-pin unbuffered SDRAM DIMMs.
Caching	
L1 Icache	16KB, direct-mapped, instruction cache on the CPU chip
L1 Dcache	8KB, direct-mapped, data cache on the CPU chip
L2 backup cache	Onboard 1MB, 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 • Cypress CY82C693U–NC chip with a PCI-to-ISA bridge, PCI EIDE, keyboard, mouse, and time-of-year clock • Two dedicated ISA expansion slots • SMC FDC37C669 combination controller chip provides control for diskettes, two UARTs with modem control, and parallel port • 1MB flash ROM
Firmware	Alpha SRM Console firmware

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

Figure 2–1 AlphaPC 164SX Switch/Connector/Component Location
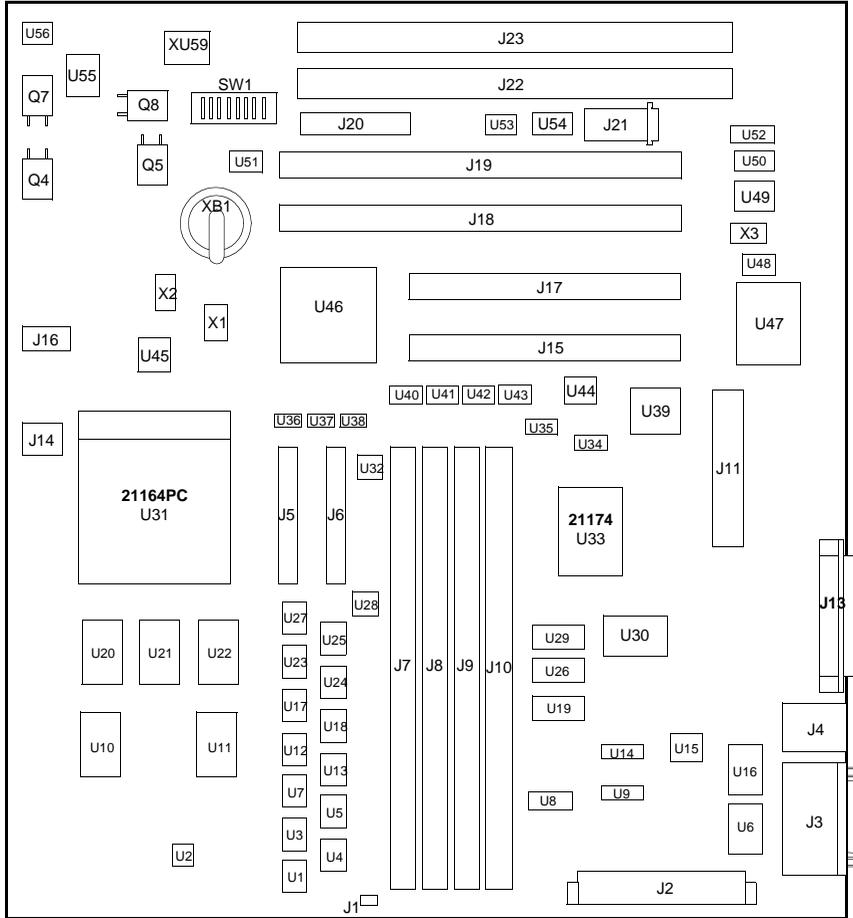
Table 2–2 AlphaPC 164SX Switch/Connector/Component List

Item No.	Description	Item No.	Description
XB1	RTC battery (CR2032)	J20	LEDs/speaker/buttons connectors
J1	Soft power connector	J21	SROM test port connector
J2	Input power (+3V, +5V, -5V, +12V, -12V)	J22	ISA slot 1
J3	COM1/COM2 (DB9) connectors	J23	ISA slot 0
J4	Keyboard/mouse connectors	SW1	Configuration switchpack
J5	EIDE drive 2/3 connector	U48	System clock PLL (CY2081)
J6	EIDE drive 0/1 connector	U10, U11, U20, U22	L2 cache data SRAMs
J7	SDRAM DIMM 0 [0:63] connector	U21	L2 cache tag SRAM
J8	SDRAM DIMM 1 [64:128] connector	U33	DIGITAL Semiconductor 21174 core logic chip
J9	SDRAM DIMM 2 [0:63] connector	U19, U26, U29	Memory address/control buffers
J10	SDRAM DIMM 3 [64:128] connector	U1, U3, U4, U5, U7, U12, U13, U17, U18, U23, U24, U25, U27	Data switches
J11	Diskette (floppy) drive connector	U40 to U43	Interrupt shift registers
J13	Parallel I/O connector	U30	Flash ROM (1MB)
J14	Microprocessor fan/fan sense connector	U46	PCI-ISA Bridge (CY82C693U-NC)
J15	PCI slot 3 (32-bit)	U36 to U38	IDE buffers
J16	Fan power, enclosure (+12V)	U47	Combination controller (FDC37C669)
J17	PCI slot 2 (32-bit)	U2, U28, U32	Reg-reg cache isolate logic
J18	PCI slot 1 (64-bit)	U31	Microprocessor, socketed (DS Alpha 21164PC)
J19	PCI slot 0 (64-bit)	U45	Microprocessor clock synthesizer (MC12439)

2.1 Power Requirements

The AlphaPC 164SX motherboard has a total power dissipation of 90 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[1]
+3.3 V dc, ±5%	5.0 A
+5 V dc, ±5%	12.0 A
–5 V dc, ±5%	0.0 A
+12 V dc, ±5%	1.0 A
–12 V dc, ±5%	100.0 mA

[1] Values indicated are for an AlphaPC 164SX motherboard with an Alpha 21164PC microprocessor operating at 400 MHz, with 128MB SDRAM, excluding adapter cards and disk drives.

Caution: **Fan sensor required**. The 21164PC microprocessor cooling fan *must* drive an RPM indicator signal. If the airflow stops, the sensor on the motherboard detects that the RPM has stopped, and resets the system.

2.2 Environmental Requirements

The 21164PC microprocessor is cooled by a small fan blowing directly into the chip's heat sink. The AlphaPC 164SX 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 164SX motherboard is specified to run within the environment listed in Table 2–4.

Table 2–4 AlphaPC 164SX 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 ±2°C/hour (20°F/hour ±4°F/hour)

2.3 Physical Parameters

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

- Length: 30.48 cm (12.0 in ±0.0005 in)

- Width: 24.38 cm (9.6 in ±0.0005 in)

- Height: 6.00 cm (2.4 in)

Physical Parameters

2.3.1 ATX Hole Specification

Figure 2–2 shows the ATX hole specification for the AlphaPC 164SX.

Figure 2–2 ATX Hole Specification
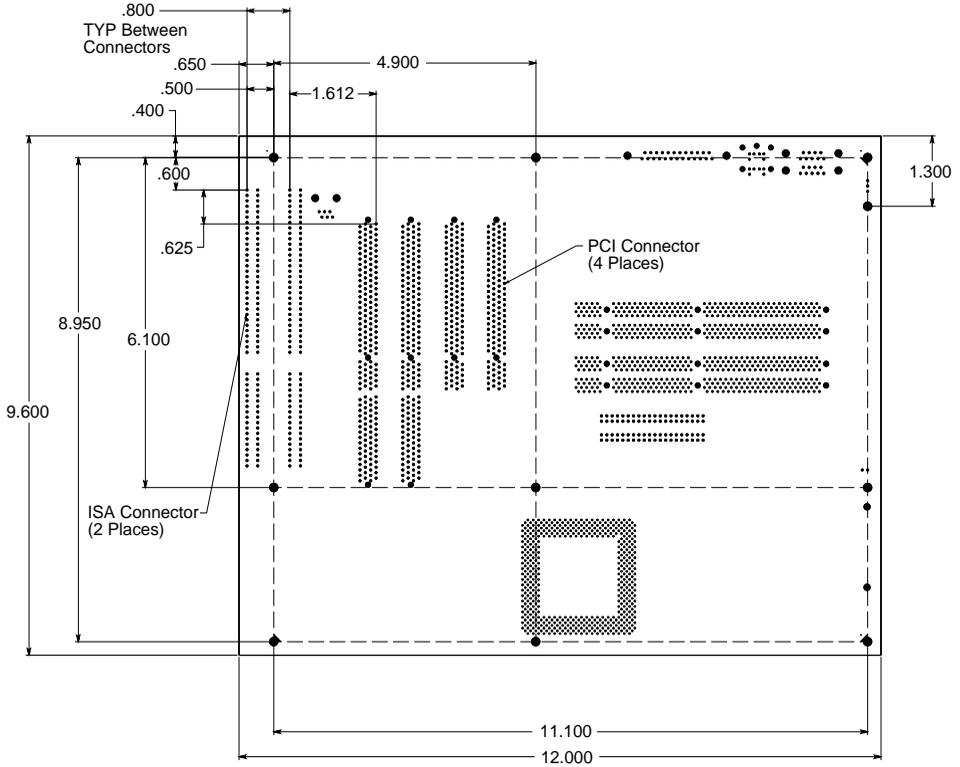
2.3.2 ATX I/O Shield Requirements

Figure 2–3 shows the ATX I/O shield dimensions for the AlphaPC 164SX.

Figure 2–3 ATX I/O Shield Dimensions

FM-05986.AI4

3
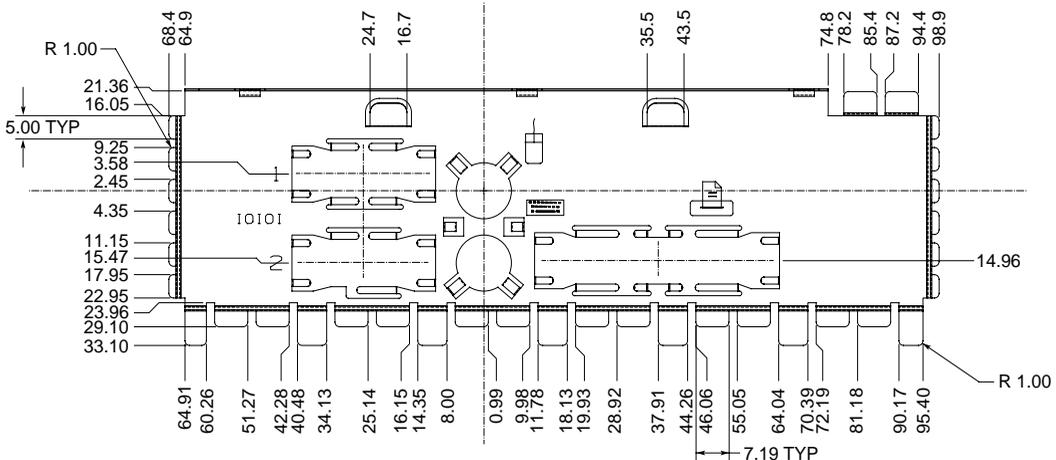
AlphaPC 164SX Switch Configuration

The AlphaPC 164SX motherboard has a switchpack located at SW1, as shown previously in Figure 2–1. These switches set the hardware configuration. Figure 3–1 shows these switch configurations.

3.1 Mini-Debugger (CF3)

The Alpha SROM Mini-Debugger is stored in the flash ROM and is enabled/ disabled by switch CF3. The default position for this switch is on (see Figure 3–1). When this switch is off, it causes the SROM initialization to trap to the Mini-Debugger after all initialization is complete, but before starting the execution of the system flash ROM code.

3.2 CPU Speed Selection (CF[6:4])

The clock synthesizer at U45 makes it possible to change the frequency of the microprocessor's clock output. The switch configuration is set in SW1, CF[6:4]. These three switches set the speed at power-up as listed in Figure 3–1. The microprocessor frequency divided by the ratio determines the system clock frequency.

3.3 Fail-Safe Booter (CF7)

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 repro-gramming the flash.

Refer to Section 8.4 for more information.

Fail-Safe Booter (CF7)

Figure 3–1 AlphaPC 164SX Configuration Switches

Note: Switch defaults are in bold.

	0	1
CF0			Reserved. **This must remain CF0=1, CF1=0**.
CF1			Reserved.
CF2			Mini-Debugger: 0 enables Mini-Debugger / **1 disables Mini-Debugger**
CF3		
CF4		
CF5		
CF6		
CF7			Fail-Safe Booter: 0: Fail Safe / **1: Alpha SRM Console**

CF Bit:	400 MHz	**533 MHz**	Note:
4	1	**1**	All other combinations are reserved.
5	1	**0**	
6	1	**1**	

4
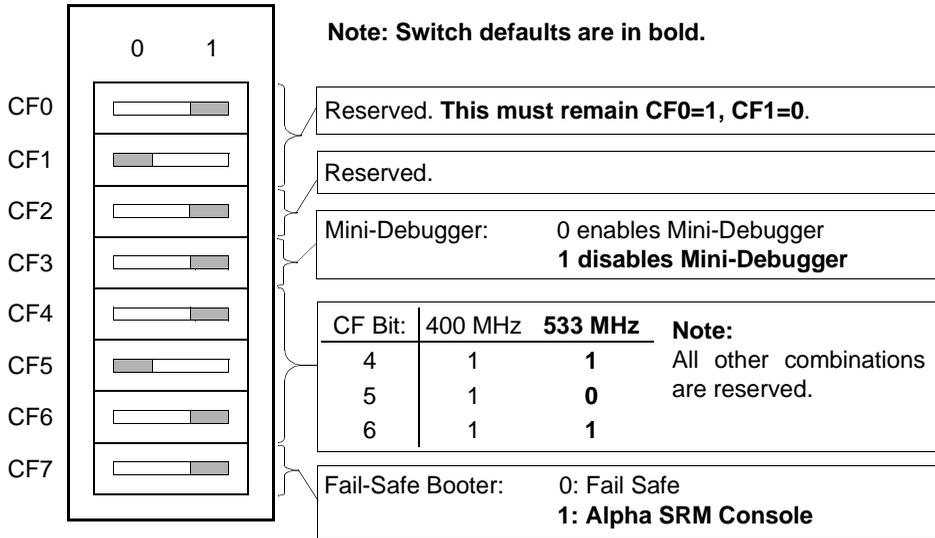
AlphaPC 164SX Connector Pinouts

This chapter lists the pinouts of the AlphaPC 164SX connectors (see Table 4–1 through Table 4–14). 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 *(Sheet 1 of 2)*

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
32-Bit and 64-Bit PCI Connectors (J15, J17, J18, J19)							
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	**PRSNT1#**	B10	—

PCI Bus Connector Pinouts

Table 4–1 PCI Bus Connector Pinouts *(Sheet 2 of 2)*

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
B11	**PRSNT2#**	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**
64-Bit PCI Connectors Only (J18, J19)							
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 (J22, J23)

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	SYSCLK	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	—	—	—	—

4.3 SDRAM DIMM Connector Pinouts

Table 4–3 shows the SDRAM DIMM connector pins.

Table 4–3 SDRAM DIMM Connector Pinouts (J7 through J10)[1]   (Sheet 1 of 2)

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

Table 4–3 SDRAM DIMM Connector Pinouts (J7 through J10)[1]   (Sheet 2 of 2)

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

[1] Pins 1 through 84 are on the front side and pins 85 through 168 are on the back side.
[2] The AlphaPC 164SX 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]**.)
[3] 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 (J5, J6)

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

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 (J11)

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 (J3)

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

4.8 Keyboard/Mouse Connector Pinouts

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

Table 4–8 Keyboard/Mouse Connector Pinouts (J4)

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

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 (J21)

Pin	Signal	Description
1	NC	—
2	**SROM_CLK_L**	Clock out
3	**Gnd**	—
4	NC	—
5	**TEST_SROM_D_L**	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 (J2)[1]

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

[1] 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 (J16)

Pin	Voltage
1	**Gnd**
2	+12 V dc
3	**Gnd**

4.12 Microprocessor Fan Power Connector Pinouts

Table 4–12 shows the microprocessor fan power connector pinouts.

Table 4–12 Microprocessor Fan Power Connector Pinouts (J14)

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

4.13 Soft Power Connector Pinouts

Table 4–13 shows the soft power connector pinouts.

Table 4–13 Soft Power Connector Pinouts (J1)

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

4.14 Multipurpose Connector Pinouts

J20 is a multipurpose connector that provides pins for the following functions:

- System speaker
- LEDs for power and the EIDE drive
- Buttons for reset and halt

Table 4–14 shows the multipurpose connector pinouts, and Figure 4–1 shows the connector layout.

Table 4–14 Multipurpose Connector Pinouts (J20) *(Sheet 1 of 2)*

Pin	Signal	Description
1	**Gnd**	—
2	**HALT_BUTTON**	Halt system
3	**Gnd**	—
4	**RESET_BUTTON**	Reset system

Multipurpose Connector Pinouts

Table 4–14 Multipurpose Connector Pinouts (J20)

Pin	Signal	Description
5	**HD_PU**	Hard drive power-up
6	**HD_LED**	Pull-up to **Vdd**
7	—	No connection
8	**POWER_LED_L**	Pull-up to **Vdd**
10, 12, 14, 16	**Gnd**	—
9	**SPKR**	Speaker output
15	**Vdd**	—
11, 13	—	No connection

Figure 4–1 Multipurpose Connector Pinouts

5
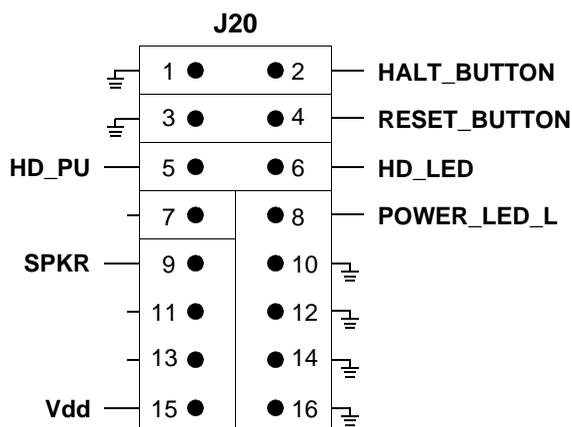
Memory and Microprocessor Configuration

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

When configuring or upgrading SDRAM, observe the following rules:

- Each DIMM must be a 168-bit unbuffered version and have a frequency of 100 MHz.

- Each bank consists of two DIMMs and must be fully populated.

- Both DIMMs in the same bank must be of equal size.

5.1 Configuring SDRAM Memory

Although not an exhaustive list, Table 5–1 lists the tested SDRAM memory configurations available. As additional configurations become available, they will be posted in online revisions of this manual on the DIGITAL Semiconductor World Wide Web Internet site. See Appendix B for the URL.

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

Refer to Figure 2–1 for DIMM connector locations.

Note: 1Mb $\times$ 72 and 1Mb $\times$ 64 DIMMs are not supported.

Table 5–1 AlphaPC 164SX SDRAM Memory Configurations

Total Memory	Bank 0 (J7 and J8)[1]	Bank 1 (J9 and J10)[1]
32MB	2Mb × 72	—
64MB	2Mb × 72	2Mb × 72
	4Mb × 72	—
96MB	4Mb × 72	2Mb × 72
128MB	4Mb × 72	4Mb × 72
	8Mb × 72	—
160MB	8Mb × 72	2Mb × 72
192MB	8Mb × 72	4Mb × 72
256MB	8Mb × 72	8Mb × 72
	16Mb × 72	—
512MB	16Mb × 72	16Mb × 72

[1] 64-bit-wide DIMMs can also be used.

5.2 Upgrading SDRAM Memory

You can upgrade memory in the AlphaPC 164SX by adding more DIMMs or replacing the ones 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. Align the DIMM so that the notch in the DIMM matches the key in the socket.

4. Firmly push the DIMM straight 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 DIGITAL Semiconductor 21164PC microprocessor with an Alpha chip that has a higher speed rating.

- Reconfigure the clock divisor switches.

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.

A DIGITAL Semiconductor 21164PC 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, 52-mm fan, 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

- 3/8-inch socket with 1/4-inch hex drive

- #2 Phillips-head screwdriver bit

5.3.3 Removing the 21164PC Microprocessor

Remove the microprocessor currently in place at location U31 by performing the following steps:

1. Unplug the fan power/sensor cable from connector J14 (see Figure 2–1).

2. Remove the four $6-32 \times 0.625$-inch screws that secure the fan to the heat sink.

3. Remove the fan.

4. If the sink/chip/fan clip is used, remove it by unhooking its ends from around the ZIF socket retainers.

5. Using a 3/8-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.

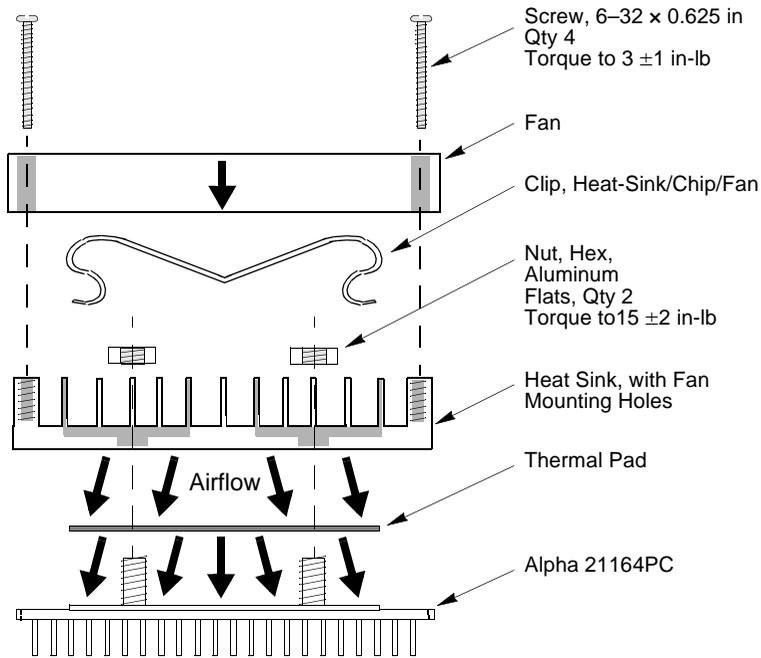10. Remove the microprocessor chip by lifting it straight out of the socket.

5.3.4 Installing the 21164PC Microprocessor

Install the new microprocessor in location U31 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.

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 Fan/Heat-Sink Assembly

Screw, 6–32 × 0.625 in
Qty 4
Torque to 3 ±1 in-lb

Fan

Clip, Heat-Sink/Chip/Fan

Nut, Hex,
Aluminum
Flats, Qty 2
Torque to15 ±2 in-lb

Heat Sink, with Fan
Mounting Holes

Thermal Pad

Alpha 21164PC

Airflow

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.

3. Place the GRAFOIL pad on the gold-plated slug surface and align it with the threaded studs.

Increasing Microprocessor Speed

    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 15 in-lb, ±2 in-lb, (2.3 Nm, ±0.2 Nm). The torque driver should have a mounted 3/8-inch socket.

        5. Insert a nut into the 3/8-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 J14 (see Figure 2–1). Fan airflow must be directed into the heat sink (fan label facing down toward the heat sink).

        2. Using a calibrated torque driver set to 3 in-lb, ±1 in-lb, secure the fan to the heat sink with four 6–32 × 0.625-inch screws.

        3. Plug the fan power/sensor cable into connector J14.

Note: When installing the microprocessor, you must change the frequency of its clock output by setting the system clock divisor switches, as described in Section 3.2.

6

Interrupts and ISA Bus Addresses

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

6.1 Interrupts

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

Table 6–1 ISA Interrupts   *(Sheet 1 of 2)*

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

Table 6–1 ISA Interrupts   *(Sheet 2 of 2)*

Interrupt Number	Interrupt Source
IRQ13	Available
IRQ14	EIDE
IRQ15	EIDE

[1] The # symbol indicates an active low signal.

6.2 ISA I/O Address Map

Table 6–2 lists the AlphaPC 164SX 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

7

Alpha SRM Console Firmware

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

7.1 Alpha SRM Console Firmware Conventions

The following conventions are used in this section:

Convention	Description
>>>	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 DIGITAL 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
```

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
-file <filename>	Specifies the name of a file to load into the system. Use the **set boot_file** command to set the environment variable that specifies a default boot file.
-flags <longword> [,<longword>]	Specifies additional information for the operating system. For DIGITAL UNIX systems, the following values may be used: i = Interactive boot s = Boot to single user a = Autoboot to multiuser Use the **set boot_osflags** command to set an environment variable that specifies a default boot flag value.
-protocols <enet_protocol>	Specifies the Ethernet protocols that will be used for a network boot. Values may be **mop** or **bootp**.
-halt	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.

Examples

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

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
gpr-name	Represents general-purpose register.
ipr-name	Represents internal processor register.
PC	Program counter.
+	The location immediately following the last location referenced by **examine** or **deposit**.

Symbolic Address	Description
–	The location immediately preceding the last location referenced by **examine** or **deposit**.
*	The location last referenced by **examine** or **deposit**.
@	The location addressed by the last location referenced by **examine** or **deposit**.

<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

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

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

examine 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.
-d	Specifies the data displayed is the decoded macro instruction. The Alpha instruction decode (-d) does not recognize machine-specific PALcode instructions.
-physical	References physical address space.
-virtual	References virtual address space.
-gpr	References general-purpose register address space.
-fpr	References floating-point register address space.

Basic Alpha SRM Console Command Descriptions

examine Command Option	Description
-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.

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.

Examples

examine Command Example	Display	Description
`>>>e r0`	`gpr:  0 (R0)  0000000000000002`	Examines the contents of R0, using a symbolic address.
`>>>e -g 0`	`gpr:  0 (R0)  0000000000000002`	Examines the contents of R0, using address space.
`>>>e grp:0`	`gpr:  0 (R0)  0000000000000002`	Examines the contents of R0, using a device name.

`>>>examine -n 5 r7`	`gpr:  38  (R7)   0000000000000000`
	`gpr:  40  (R8)   0000000000000000`
	`gpr:  48  (R9)   0000000000000000`
	`gpr:  50  (R10)  000000007FFBF800`
	`gpr:  58  (R11)  000000007FF781A2`
	`gpr:  60  (R12)  0000000000000000`
		Examines the contents of R7 and the next five registers.

`>>>examine ipr:11`	`ipr  11  (KSP)  FFFFFFFF8228DFD0`	Examines the contents of internal processor register 11.

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

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

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

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
`config`	Displays the current memory configuration, PCI logical slots, and ISA logical slots.
`device` [device name]	Displays the devices and controllers in the system. Specifying a device name returns information on that device only.
`iobq`	Displays the input/output counter blocks.
`hwrpb`	Displays the hardware restart parameter block.
`map`	Displays the system virtual memory map.
`memory`	Displays the memory module configuration.
`pal`	Displays the version of DIGITAL UNIX PALcode.
`version`	Displays the version of the console.
`<envar>`	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.

Examples

show Command Example	Description
>>>**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	Lists device information, such as system designation, drive model, or Ethernet address.
>>>**show memory** 48 Meg of System Memory	Lists system random-access memory (RAM) size.
>>>**show *** (refer to Section 7.3)	Lists all environment variables and their settings.
>>>**show boot*** (refer to Section 7.3)	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
`auto_action`	When used with the **set** or **show** 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.
`boot_file`	When used with the **set** or **show** command, this variable modifies or displays the file name to be used when a bootstrap requires a file name. The default is null.
`boot_osflags`	When used with the **set** or **show** command, this variable modifies or displays the additional parameters to be passed to system software. The default is 0.
`bootdef_dev`	When used with the **set** or **show** 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
console	When used with the **set** command, this variable modifies the console output to either the serial port or the graphics controller.
control_scsi_term	This variable is unused in the motherboard system.
ewa0_mode	This variable determines if the AUI (ThinWire) or the twisted-pair Ethernet ports will be enabled. AUI is the default. (Autosensing is not supported.)
os_type	When used with the **set** or **show** command, this variable modifies or displays the specified firmware that will be loaded on the next power cycle. Specify the value **osf** or **UNIX** to select the Alpha SRM Console.
pci_parity	This variable controls PCI parity checking. The possible values are: **on** = Parity checking is enabled. **off** = Parity checking is disabled; this is the default. **sniff** = Parity checking is enabled or disabled depending on the PCI device.
oem_string	When used with the **set** or **show** 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

language n	The language environment variable assigns language n to the system (where n is the option number of a language listed in the menu that follows). Use the following procedure to select the language:
	
	1. At the Alpha SRM Console prompt, enter the following commands:
	
	    `>>>set language 0`
	    `>>>init`
	
	The following menu and prompt are displayed:
	
	`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)`
	
	`(1..16):`
	
	2. 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:
	
	`(1..16):36`
	
	3. When you receive a message to reset the system, power cycle the system.

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.

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 compo-
nents, 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
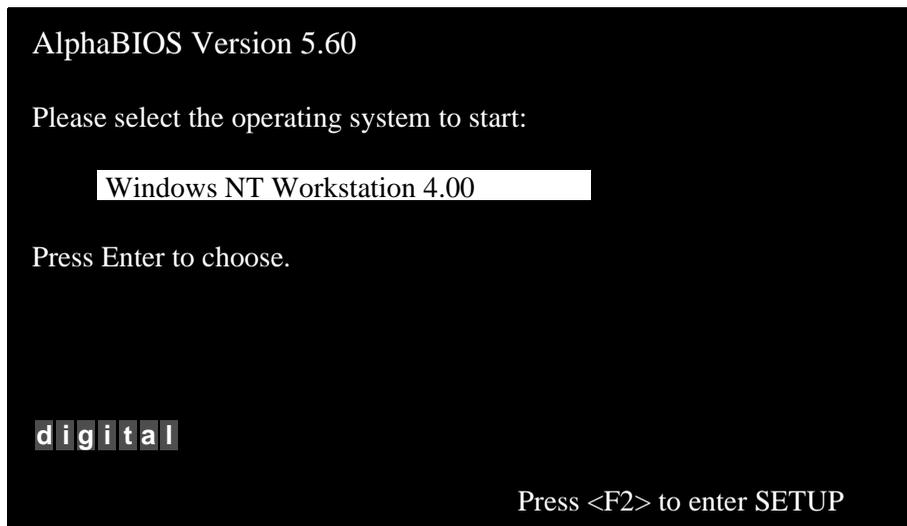
3. Press **F2** to start the AlphaBIOS setup program.

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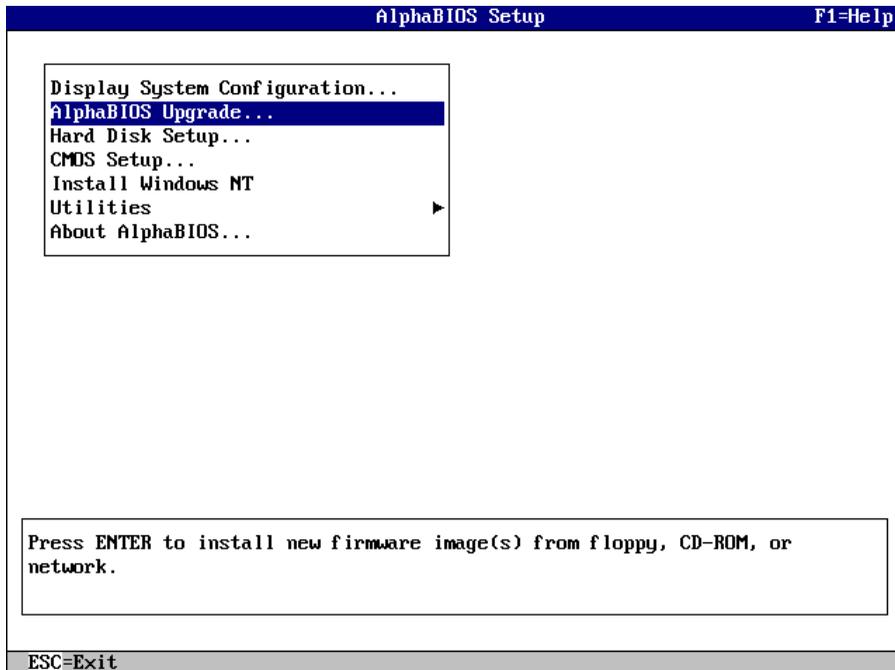
Press **Enter** to begin installing the SRM Console firmware image.

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
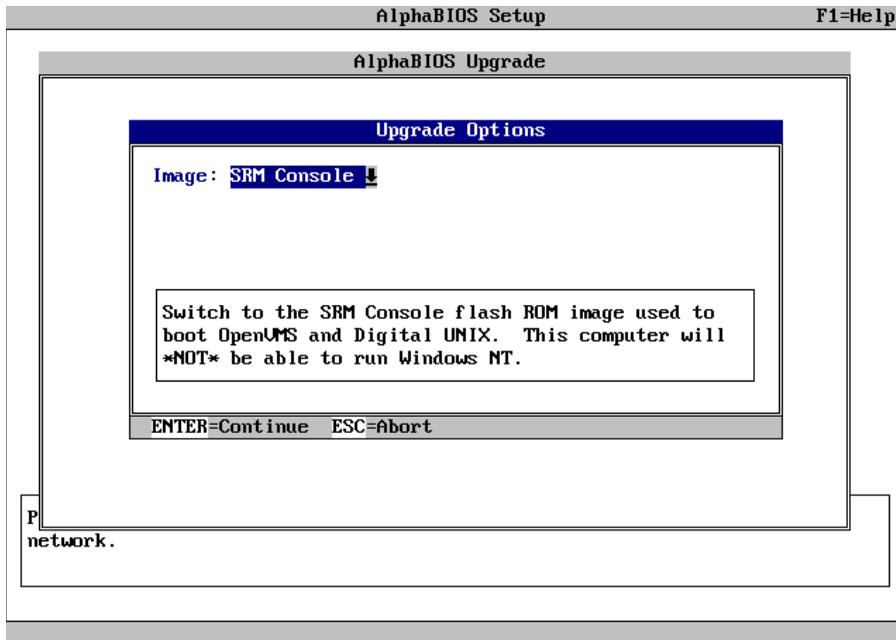
```
                        AlphaBIOS Setup                           F1=Help

                         AlphaBIOS Upgrade


                            Upgrade Options

        Image:  SRM Console ⬇



            Switch to the SRM Console flash ROM image used to
            boot OpenVMS and Digital UNIX.   This computer will
            *NOT* be able to run Windows NT.


         ENTER=Continue   ESC=Abort


    P
    network.
```

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

```
                          AlphaBIOS Setup                        F1=Help

                         AlphaBIOS Upgrade


                          Upgrade Options

        Image: SRM Console ▼
                            WARNING
                  You have selected to switch to a
                  different operating system.  This
                  computer will no longer be able
          Swi   to run Windows NT.  Are you       age used to
          boo   *REALLY* sure you want to do       omputer will
          *NO   this?

                  ENTER=Continue   ESC=Back



        P
       network.


```

Press **Enter** to continue the installation.

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

```
                          AlphaBIOS Setup                        F1=Help

                           AlphaBIOS Upgrade

                          Upgrade SRM Console

        Old Version:     AlphaBIOS 5.60 970430.1740
        New Version:     SRM Console


        WARNING: This will take several seconds.
                 Do *NOT* reset or power-cycle the system before
                 receiving notification that the upgrade process
                 has completed.


        F10=Continue   ESC=Abort

    P
  network.

```
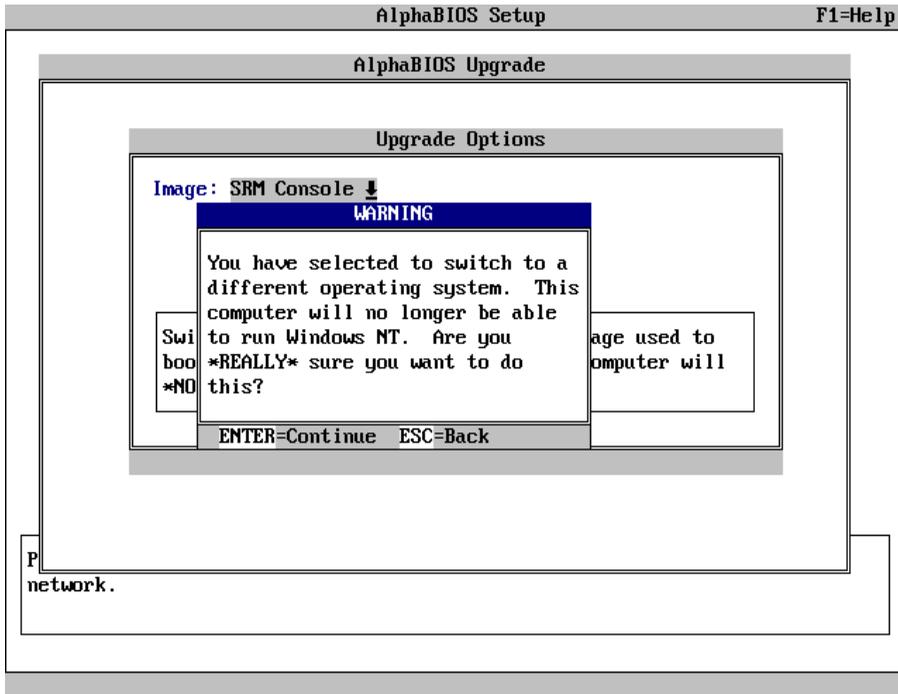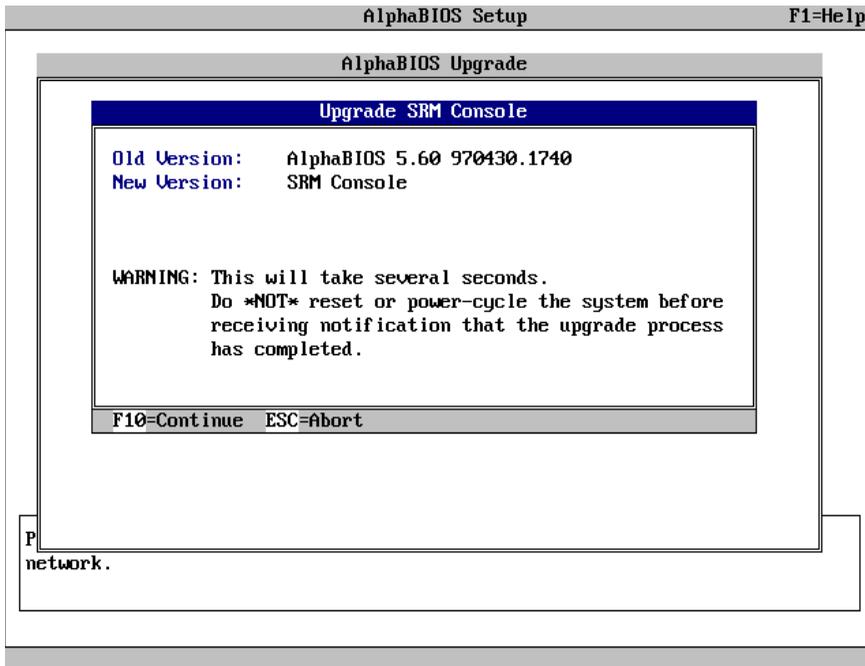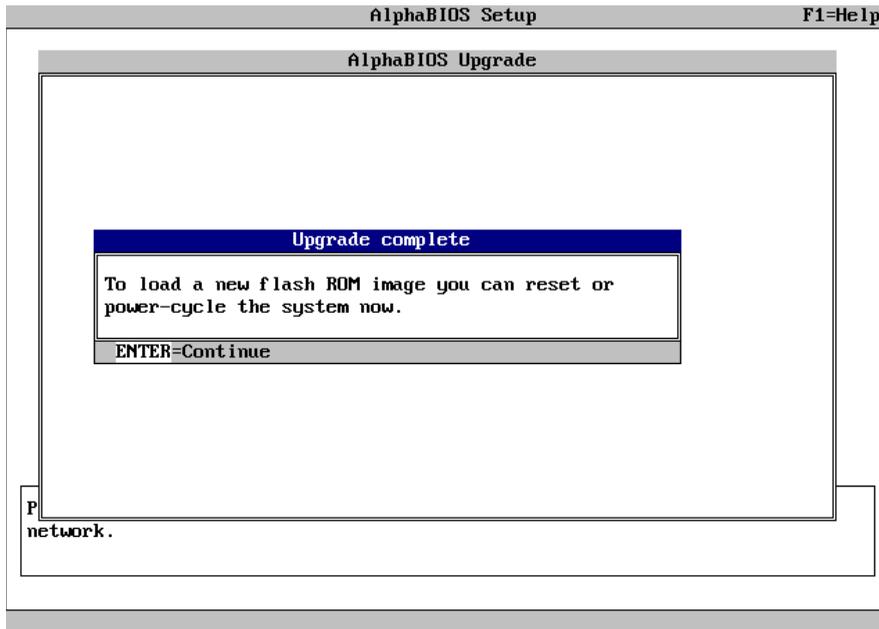
Press **F10** to continue the installation.

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

```
                          AlphaBIOS Setup                         F1=Help
                          AlphaBIOS Upgrade



                          Upgrade complete
            To load a new flash ROM image you can reset or
            power-cycle the system now.

              ENTER=Continue




          P
          network.
```

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

7.5 Installing the DIGITAL UNIX Operating System

Note: DIGITAL UNIX Version 4.0D is the minimum revision required for the AlphaPC 164SX motherboard system.

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

Note: If you already have a previous version of DIGITAL 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 DIGITAL UNIX operating system on an AlphaPC 164SX 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

- DIGITAL UNIX Version 4.0D or higher Operating System Volume 1 compact disc

- Alpha SRM Console Version 4.9 or higher

8
Troubleshooting

This chapter contains information about troubleshooting hardware and software during AlphaPC 164SX 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/switch selection matches the speed of the Alpha chip.

4. Check that the CPU fan is connected and spinning.

5. 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.

6. Reseat the video card and ensure that it is connected to the monitor.

7. Reseat the DIMMs.

8. Replace the DIMMs.

8.2 Beep Codes

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

Table 8–1 Beep Codes

Beep Code	Description
1–2–3[1]	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.

[1] 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   *(Sheet 1 of 3)*

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
	0C	Memory sized and memory bank 0 written
	0D	Enable Dcache
	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
	20	ISA bus reset

Table 8–2 Post Codes *(Sheet 2 of 3)*

Source	Post Code (hex)	Description
	3A	Jump to Mini-Debugger
	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	FD	I/O bus initialization
	FC	I/O initialization complete
	FB	UART initialization

Table 8–2 Post Codes   *(Sheet 3 of 3)*

Source	Post Code (hex)	Description
	BF	Fail-safe booter is scanning the diskette
	1	Fail-safe booter is loading the AlphaBIOS
	2	Fail-safe booter is loading the Debug Monitor
	3	Fail-safe booter is loading the SRM console
	4	Fail-safe booter is loading the Linux mini-loader

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 164SX 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. Slide switch CF7 to off, as described in Section 3.3.

  3. Power on your system.

  4. Insert the *AlphaPC 164SX SRM Console Firmware* diskette into the diskette drive.

  5. Proceed to Section 7.4 and follow the procedures.

9
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
Supporting Products

This appendix lists sources for components and accessories that are not included with the AlphaPC 164SX. For the latest information, visit the Alpha website at URL: **http://www.alpha.digital.com**. Click on **Motherboard Products**. The hardware compatibility list (HCL) is also available at this location.

A.1 Memory

Dual inline memory modules (DIMMs) are available from:

Micron Semiconductor Products, Inc.
8000 South Federal Way
Mail Stop 607
Boise, ID 83706
Phone: 208-368-3900
Fax: 208-368-5018
PN MT9LSDT272AG-66CL2 (2Mb × 72)
PN MT18LSDT472AG-66CL2 (4Mb × 72)

Dataram
P.O. Box 7528
Princeton, NJ 08543
Phone: 1-800-DATARAM (328-2726), ext. 239
PN 60057 (8Mb × 72)
PN 60061 (16Mb × 72)

Memory

Viking Components
attn: Carl Johnson, OEM Development Executive
30200 Avenida de la Banderas
Rancho Santa Margarita, CA 92688
Phone: 1-800-338-2361, ext. 316
Fax: 714-643-7250
PN VE2641U4SN3-DC01 (2Mb $\times$ 64)
PN VE8641U4SN3-DC01 (8Mb $\times$ 64)
PN VE16642U4SN3-DC01 (16Mb $\times$ 64)
PN VE2721U4SN3-DC01 (2Mb $\times$ 72)
PN VE4721U4SN3-DC01 (4Mb $\times$ 72)
PN VE8721U4SN3-DC01 (8Mb $\times$ 72)
PN VE16722U4SN3-DC01 (16Mb $\times$ 72)

NEC Electronics, Inc.
The Meadows Building, 4th Floor
161 Worcester Road
Framingham, MA 01701
Phone: 508-935-2000
Fax: 508-935-2233
PN MC-452AA724F-A10 (2Mb $\times$ 72)
PN MC-454AC724F-A10 (4Mb $\times$ 72)
PN MC-458AA724F-A10 (two-bank, 8Mb $\times$ 72)
PN MC-4516AC724F-A10 (two-bank, 16Mb $\times$ 72)
PN MC-458CA724F-A10 (four-bank, 8Mb $\times$ 72)
PN MC-4516CC724F-A10 (four-bank, 16Mb $\times$ 72)

Samsung America, Inc.
3655 N. First Street
San Jose, CA 95134
Phone: 1-800-423-7364
PN KM374S203ATN-G0 (2Mb $\times$ 72)
PN KM374S403ATN-G0 (4Mb $\times$ 72)

Kingston Technology Company
17600 Newhope Street
Fountain Valley, CA 92708
Phone: 1-800-845-2545
PN KTV164SX/16 (2Mb $\times$ 64)
PN KTV164SX/32 (4Mb $\times$ 64)
PN KTV164SX/64 (8Mb $\times$ 64)
PN KTV164SX/128 (16Mb $\times$ 64)
PN KTV164LX/16 (2Mb $\times$ 72)
PN KTV164LX/32 (4Mb $\times$ 72)
PN KTV164LX/64 (8Mb $\times$ 72)
PN KTV164LX/128 (16Mb $\times$ 72)

A.2 Thermal Products

Components included in this heat-sink and fan solution are heat sink, GRAFOIL
pad, two hex nuts, heat-sink clips, 52-mm fan, and four screws. These are available
from:

United Machine and Tool Design
River Road
Fremont, NH 03044
Phone: 603-642-5040
Fax: 603-642-5819
PN 70-33148-01

A.3 Enclosure

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

Axxion Group Corporation
7801 Trade Center Avenue
El Paso, TX 79912
Phone: 915-877-5288
PN DL17

A.4 Power Supply

An ATX form-factor power supply, suitable for use with the AlphaPC 164SX (+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 AP2-5300FRV (300 W)

Antec, Inc.
2859 Bayview Drive
Fremont, CA 94538
Phone: 510-770-1200, ext. 312
PN PP-253V (250 W)

B

Associated Documentation

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

Title	Vendor
Alpha AXP Architecture Reference Manual PN EY–T132E–DP	Call your local distributor or call Butterworth-Heinemann (Digital Press) at 1-800-366-2665
Alpha Architecture Handbook[1] PN EC–QD2KB–TE	See previous entry
DIGITAL Semiconductor Alpha 21164PC Microprocessor Hardware Reference Manual PN EC–R2W0A–TE	Digital Equipment Corporation DIGITAL Semiconductor 77 Reed Road Hudson, MA 01749 USA Online, World Wide Web access at URL: **http://www.digital.com/semiconductor**
DIGITAL Semiconductor Alpha 21164PC Microprocessor Data Sheet PN EC–R2W1A–TE	See previous entry
DIGITAL Semiconductor 21174 Core Logic Chip Technical Reference Manual PN EC–R12GB–TE	See previous entry
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 1.0 PCI BIOS Specification, Revision 2.1	PCI Special Interest Group U.S. 1–800–433–5177 International 1–503–797–4207 Fax 1–503–234–6762

Title	Vendor
CY82C693U hyperCache/Stand-Alone PCI Peripheral Controller with USB Data Sheet	Cypress Semiconductor Corporation 3901 North First Street San Jose, CA 95134 Phone: 1-800-858-1810
Super I/O Floppy Disk Controller with Infrared Support (FDC37C669) Data Sheet	Standard Microsystems Corporation 80 Arkay Drive Hauppauge, NY 11788 Phone: 1-800-443-7364 Fax: 1-516-231-6004

[1] This handbook provides information subsequent to the *Alpha AXP Architecture Reference Manual*.