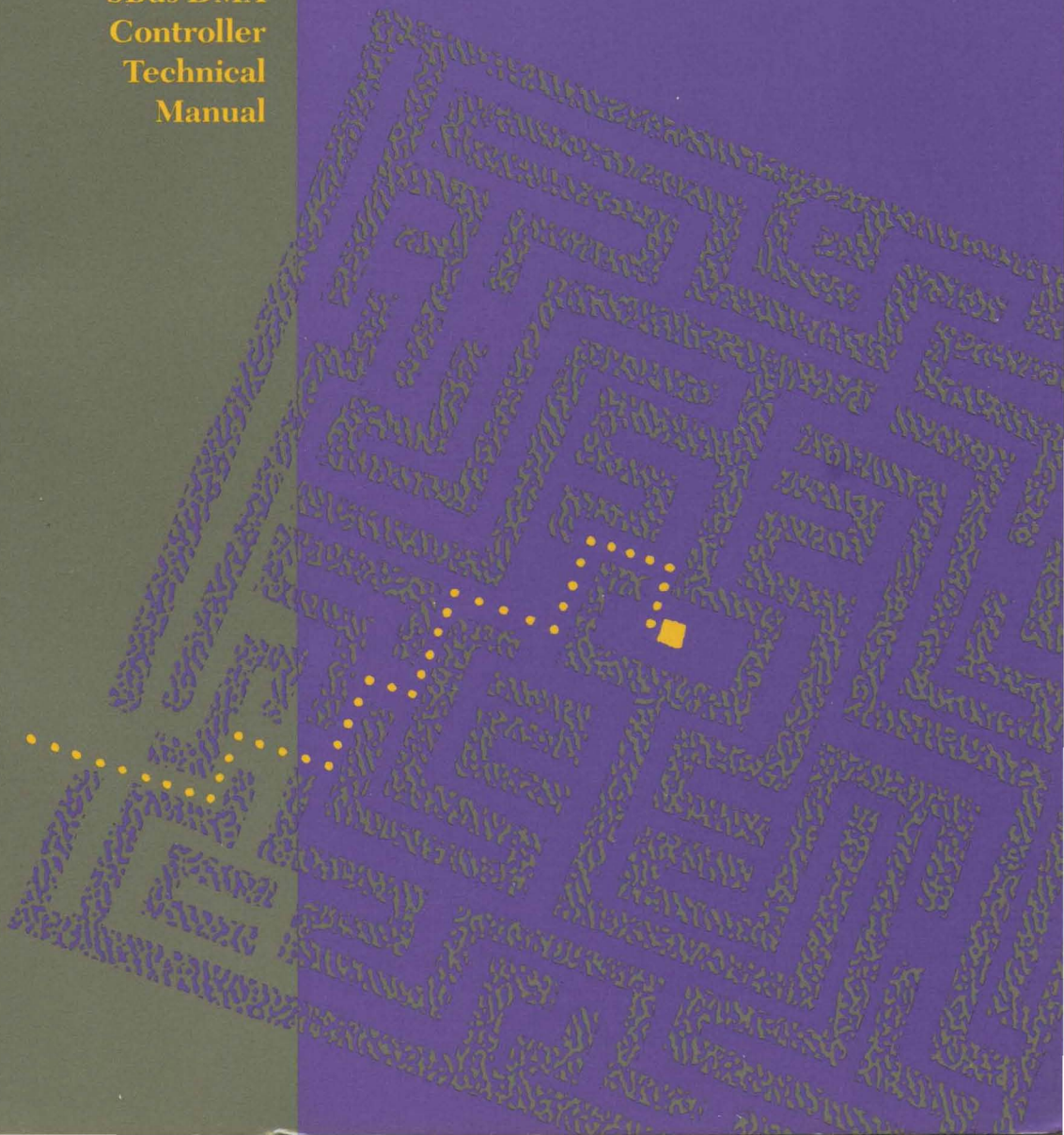
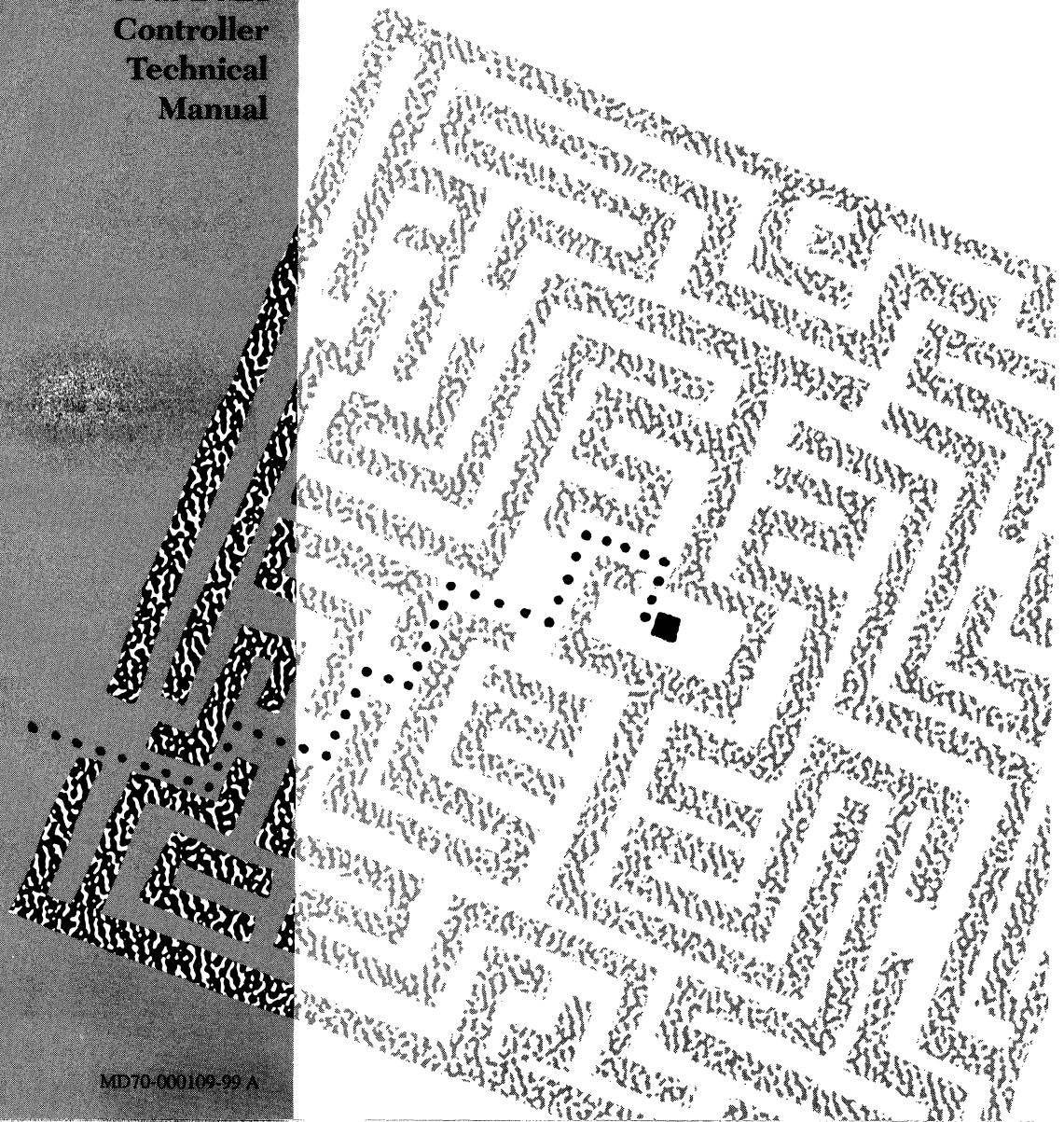
LSI LOGIC

L64853 SBus DMA Controller Technical Manual



LSI LOGIC

L64853 SBus DMA Controller Technical Manual



MD70-000109-99 A

Publications are stocked at the address given below. Requests should be addressed to:

LSI Logic Corporation
1551 McCarthy Boulevard
Milpitas, CA 95035
Fax 408.433.6802

This document is derived in part from documents created by Sun Microsystems and thus constitutes a derivative work.

TRADEMARK ACKNOWLEDGMENT

LSI Logic and the logo design are trademarks of LSI Logic Corporation.

Sun and SPARC are trademarks of Sun Microsystems, Inc.

The L64853 SBus DMA Controller Technical Manual is written for two audiences: system-level programmers and hardware designers. The manual assumes readers are familiar with computer architecture, software and hardware design, and design implementation. It also assumes readers have access to additional information about the SPARC workstation—in particular, the SPARC SBus specification, which can be obtained from Sun Microsystems.

This manual is organized in a top-down sequence; that is, the earlier chapters describe the purpose, context, and functioning of the L64853 DMA Controller from an architectural perspective, and later chapters provide implementation details.

Content and Organization for System-Level Programmers—For the system-level programmer, this manual briefly describes the SPARC workstation architecture, then fully describes the function of the chip with respect to the architecture.

Chapter 1 describes the basic features and operation of the L64853 DMA Controller and the chip's relationship to the other SPARC chips. Chapter 2 describes the internal, programmable registers of the L64853, and it describes how to access both internal and external registers. Chapter 3 further describes the operation of the E and D channels. To aid in implementation of the chip in a typical environment, Chapter 6 shows how to access the Emulex SCSI Processor (ESP-100) and the AMD LANCE Ethernet AM7900 internal registers.

Content and Organization for Hardware Designers—For the hardware designer, this manual provides the electrical, logical, and mechanical data necessary to integrate the L64853 DMA Controller into the SPARC workstation.

Chapter 4 describes the data transfer cycles for the D and E channels and the signals used by the L64853 DMA Controller during read and write operations. Chapter 5 provides the AC, DC, environmental, and mechanical specifications for the L64853.

This manual is divided into six chapters:

- **Chapter 1: Introduction** describes the basic features and operation of the L64853 DMA Controller, showing its relationship to the other SPARC system elements.
- **Chapter 2: Registers** describes the types of registers available on the L64853 DMA Controller and how internal and external registers are accessed.
- **Chapter 3: Operation of the D and E Channels** describes the DMA Controller's D and E channels, which are used with peripheral chips, and modes of operation on the Sbus, which is used for data transfers between the peripheral subsystems and main memory.
- **Chapter 4: Interface Description** describes the signals used by the L64853 and the data transfer cycles for the D and E channels.
- **Chapter 5: Specifications** provides the AC, DC, environmental, and mechanical specifications for the L64853.

- **Chapter 6: Applications** describes the types of SPARC workstation applications that the L64853 DMA Controller fits into and provides interface examples to Emulex SCSI Processor (ESP-100) and to an AMD Local Area Network Controller (AM7990).

Related Publications

The SBus Specification, Revision A (September 1989)
Sun Microsystems, 2550 Garcia Avenue, Mountain View, CA 94042

Chapter 1: Introduction **1-1**
1.1 General Description of the L64853 SBus DMA Controller 1-1
• Architecture 1-2
• Channel Operations 1-3
1.2 Key Features 1-3

Chapter 2: Registers **2-1**
2.1 Overview of L64853 Internal Registers 2-1
2.2 Internal Programmable Registers 2-1
• DMA Control/Status Register 2-2
• DMA Address Counter 2-4
• DMA Byte Counter 2-4
• Internal ID Register 2-5
2.3 Addressing Internal and External Registers 2-5

Chapter 3: Operation of the D and E Channels **3-1**
3.1 Modes of Operation 3-1
3.2 DMA Reads and Writes (Master Mode) 3-1
3.3 Register Reads and Writes (Slave Mode) 3-3
3.4 Specifics of D Channel Operation 3-4
• Termination and Interrupts 3-5
• Use of the Byte Counter Register 3-5
3.5 Specifics of E Channel Operation 3-6

Chapter 4: Interface Description **4-1**
4.1 Pin Summary 4-1
4.2 Signal Descriptions 4-3
• D Channel Signals 4-3
• E Channel Signals 4-4
• SBus Interface Signals 4-6
• Miscellaneous Signals 4-8
4.3 Data Transfer Cycles 4-8

Chapter 5: Specifications **5-1**
5.1 AC Characteristics 5-1
5.2 DC Characteristics 5-8

Chapter 6: Applications **6-1**
6.1 The SBus and the SPARC Workstation 6-1
6.2 Using the L64853 With the EMULEX ESP100 SCSI Processor 6-2
• Electrical Interface 6-2
• Accessing ESP Internal Registers 6-4
6.3 Using the L64853 (With the AMD LANCE Ethernet Controller) 6-5
• Electrical Interface 6-5
• Accessing LANCE Internal Registers 6-6

Figures

Figure 1.1 L64853 Logic Symbol 1-2
Figure 1.2 L64853 Internal Block Diagram 1-2
Figure 2.1 Control/Status Register (DMA_CSR) 2-2
Figure 2.2 DMA Address Counter (DMA_ADDR) 2-4
Figure 2.3 DMA Byte Counter (DMA_BCNT 2-4
Figure 2.4 ID Register 2-5
Figure 4.1 L64853 Pin Diagram (Top View) 4-1
Figure 4.2 SBus DMA Read/Write 4-9
Figure 4.3 E Channel DMA Read Cycle (Data not in Pack/Unpack Register) 4-10
Figure 4.4 E Channel DMA Read Cycle (Data in Pack/Unpack Register) 4-11
Figure 4.5 E Channel Register Read/Write Cycles 4-12
Figure 4.6 D Channel DMA Read—Fast Cycle 4-12
Figure 4.7 D Channel DMA Write—Fast Cycle 4-13
Figure 4.8 D Channel Register Read—Fast Cycle 4-13
Figure 4.9 D Channel Register Write—Fast Cycle 4-14
Figure 4.10 Extended D Channel DMA Read—Slow Cycle 4-14
Figure 4.11 Extended D Channel DMA Write—Slow Cycle 4-15
Figure 4.12 Extended D Channel DMA Register Read—Slow Cycle 4-15
Figure 4.13 Extended D Channel DMA Register Write—Slow Cycle 4-16
Figure 4.14 Offboard ID Read—Fast Cycle 4-16
Figure 5.1 SBus Input Signals 5-4
Figure 5.2 SBus Output Signals 5-5
Figure 5.3 DMA Data Bus Read/Write Cycle Timing (Fast Cycle) 5-5
Figure 5.4 DMA Data Bus DMA Cycle Timing (Fast Cycle) 5-6
Figure 5.5 LANCE Read/Write Cycle Timing 5-6
Figure 5.6 LANCE DMA Read/Write Cycle Timing 5-7
Figure 5.7 120-Pin PQFP Dimension Drawing 5-10
Figure 6.1 Host-Based System: SPARCstation 1 6-1
Figure 6.2 Symmetric Configuration 6-2
Figure 6.3 L64853 to ESP100 Interface 6-3
Figure 6.4 ESP Interface Example Using External Boot PROM 6-4
Figure 6.5 L64853 Am7990 LANCE Interface 6-6

Tables

Table 2.1 Accesses by Register Type 2-5
Table 2.2 Internal Register Addresses 2-6
Table 4.1 Signal Summary 4-2
Table 5.1 Signal Propagation Delays 5-1
Table 5.2 L64853 Switching Characteristics (Part 1 of 3) 5-2
Table 5.2 L64853 Switching Characteristics (Part 2 of 3) 5-3
Table 5.2 L64853 Switching Characteristics (Part 3 of 3) 5-4
Table 5.3 Absolute Maximum Ratings 5-8
Table 5.4 Recommended Operating Conditions 5-8
Table 5.5 DC Characteristics 5-9
Table 6.1 Physical Address Line Wiring on SBus for Selecting External Registers on ESP SCSI Chip 6-5
Table 6.2 Wiring of SBus Physical Address Lines for Selecting External Registers on LANCE ETHERNET Chip 6-6

Chapter 1: Introduction

This chapter describes the basic capabilities and operation of the L64853 DMA Controller, showing its relationship to the other SPARC-system components. Its intended readers include system-level programmers and hardware designers, for whom this chapter is particularly important.

1.1 General Description of the L64853 SBus DMA Controller

The L64853 SBus DMA Controller provides a complete SBus interface for SBus peripheral subsystems. The L64853 provides two independent DMA channels, a 16-bit channel and an 8-bit channel. The L64853, implemented in a 1.5-micron CMOS process and manufactured by LSI Logic Corporation, is packaged in an inexpensive, 120-pin, plastic quad flat package (PQFP).

The L64853's two channels support DMA for use in applications that require operation as an SBus master. Such applications include:

- Ethernet controllers
- 8-bit controllers

The L64853's two channels can also be used for applications that rely on programmed I/O and thus use only the L64853's SBus slave capability. Such applications include:

- Serial ports
- Analog-to-Digital converters

The L64853's 8-bit channel is called the D channel. The 16-bit channel is called the E channel. The L64853 generates, upon request from devices attached to either channel, sequences of SBus data transfers (that is, reads or writes) between the peripheral controller and main memory. To perform this function, the L64853 can become an SBus master through the use of the SBus Request and Grant signals. (In SBus terminology, the L64853 is a "DVMA Master" that is, it generates virtual addresses on the SBus data lines and employs the SBus controller's Memory Management Unit (MMU) to translate these virtual addresses into physical addresses. (See the *Sun SBus Specification* from Sun Microsystems for more details.)

The L64853 is programmable. Programs running on the system CPU may set parameters that govern the transfers performed by the L64853 and also read the current status of the chip. This programming is achieved by writing and reading internal registers on the L64853 (and also on the two peripheral controllers). In order to permit this writing and reading, the L64853 can become an *SBus slave* when the CPU asserts the Slave Select signal specific to the L64853. While acting as an SBus Slave, the L64853 can also be used by the CPU as a conduit to the two peripheral controller chips, thus allowing the CPU to program them.

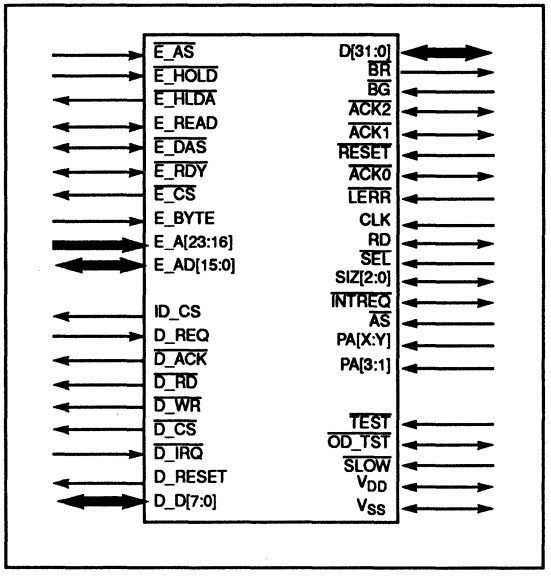


Figure 1.1 L64853 Logic Symbol

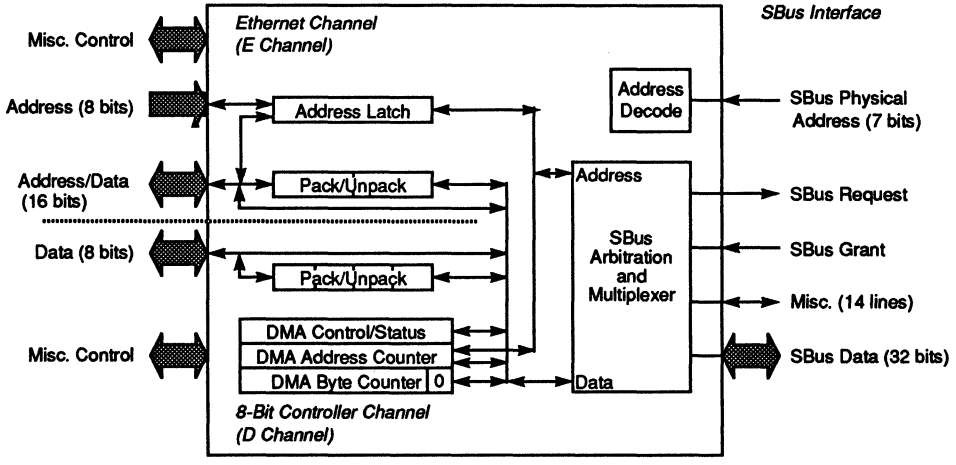


Figure 1.2 L64853 Internal Block Diagram

Architecture

The principal components of the L64853 are its two functionally distinct DMA channels and its SBus interface with associated bus arbitration logic. Figure 1.1, illustrating the L64853 signals, reflects this organization, as does Figure 1.2, which shows the chip's internal architecture. Chapter 4 describes the L64853 DMA Controller's signals.

The architecture of the chip can be summarized by examining its registers. The E channel ("E" channel) has three registers:

- An Address Latch Register for temporarily storing memory addresses from the E channel controller before they are put on the SBus
- A Pack Register and an Unpack Register for buffering 32-bit SBus words into bytes or 16-bit half-words

The 8-bit I/O controller channel ("D" channel) has five registers:

- A Pack Register and an Unpack Register for buffering 8-bit transfers from the peripheral controller into 32-bit transfers on the SBus
- A Control/Status Register containing programmable parameters and status fields
- An Address Count Register for holding the memory address of the current word being transferred on the SBus
- A Byte Count Register for counting down the number of bytes transferred during a transfer of a block of data

Channel Operations

As mentioned, the L64853 becomes an SBus master to perform reads and writes between the peripheral controllers and main memory. Alternatively, the L64853 becomes an SBus slave, in which mode register reads and writes are performed. All channel operations are ultimately governed by the SPARC environment in which the L64853 is implemented. Section 1.2 and Chapter 3 describe the most general-case use of the L64853. Later chapters on signals and timing specifications provide detailed information on AC and DC characteristics. Examples of using the AMD LANCE controller and the Emulex ESP-100 appear in Chapter 6.

1.2 Key Features

As its key features, the L64853 offers:

- Support for an 8- or 16-bit peripherals
- Packing/unpacking of 32-bit SBus words into bytes or 16-bit half-words for use by the peripheral controllers
- Support for byte, half-word, or word transfers on the SBus
- Operation in virtual address space with the SPARC MMU providing logical-to-physical address translations
- Support for rerun acknowledgments
- Inclusion of 24-bit address and data counters
- Packaging in a low-cost 120-pin plastic flat pack

- Use of a single clock input

These features are described in more detail in subsequent chapters.

Chapter 2: Registers

This chapter first discusses the kinds of registers available on the L64853, then describes the chip's programmable and ID registers, and ends by telling how to access both internal and external registers. It is intended primarily for system programmers.

2.1 Overview of L64853 Internal Registers

The L64853 has three types of internal registers:

1. Programmable registers

   The L64853's three *programmable registers* are used to control the operation of the chip and to contain status information. They are the following:

   - DMA Control/Status Register
   - DMA Address Counter
   - DMA Byte Counter.

2. ID Register

   The *ID Register* is pre-programmed with a read-only value unique to the L64853 chip.

3. Pack and Unpack Registers

   Both the D and the E channel have a *Pack Register* and an *Unpack Register*. These registers are used to buffer data transfers between the 32-bit SBus and the 8-bit D channel or the 16-bit E channel.

Because the operation of the Pack/Unpack Registers is entirely automatic, it is described only in the context of channel operations (Chapter 3). The rest of this section focuses on the programmable registers and the ID Register.

2.2 Internal Programmable Registers

The three internal programmable L64853 registers are defined in this section. Chapter 3 provides additional details about field and bit operations of these registers while the D and E channels are operating.

These registers contain control and status information. The internal programmable L64853 registers are used solely to control the D channel and the SBus. In contrast, the E channel is controlled entirely by external registers on the E peripheral controller itself.

DMA Control/Status Register

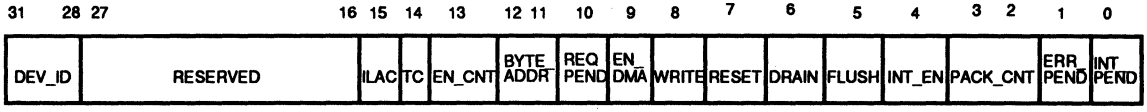


Figure 2.1 Control/Status Register (DMA_CSR)

The 32-bit Control/Status Register (shown in Figure 2.1 above) directs the functioning of the D channel interface and also reports the status of that channel. The functions of the individual fields are given in the following description. The set state = 1, the clear state = 0.

INT_PEND **Interrupt Pending** **Bit 0 (Read Only)**

This bit is set automatically to indicate a pending interrupt (that is, whenever TC is set or when $\overline{\text{D\_IRQ}}$ is asserted); otherwise, it is clear.

NOTE: Throughout this manual, a signal that is asserted when its voltage is low is indicated by an overbar; for example, $\overline{\text{D\_IRQ}}$ refers to the D channel DMA INTERRUPT REQUEST, active low signal. The underbar is a connective to the second part of the signal name.

ERR_PEND **Error Pending** **Bit 1 (Read Only)**

This bit is set when an error condition occurs during a D channel memory access (for example, a parity error, protection violation, or timeout). ERR_PEND causes an interrupt ($\overline{\text{INTREQ}}$ asserted) if INT_EN is set. DMA transfers over the SBus are stopped when this condition occurs. ERR_PEND is reset whenever FLUSH is set.

PACK_CNT **Pack Count** **Bits 3-2 (Read Only)**

This field contains the number of bytes in the D channel Pack/Unpack Register. The field is cleared by setting FLUSH or DRAIN; however, FLUSH merely clears the field, whereas DRAIN writes the contents of the Pack/Unpack Register to memory and then clears the field.

INT_EN **Interrupt Enable** **Bit 4 (Read/Write)**

When set, this bit enables the passing through of D channel interrupts ($\overline{\text{D\_IRQ}}$ to SBus $\overline{\text{INTREQ}}$), as well as passing through interrupts from ERR_PEND and TC.

FLUSH **Flush Buffer** **Bit 5 (Write Only)**

When set, this bit causes PACK_CNT, ERR_PEND, and TC to be reset. This bit always reads as clear.

DRAIN **Drain Buffer** **Bit 6 (Read/Write)**

When set, this bit causes any bytes yet in the D channel Pack Register to be written out to SBus memory; then it clears PACK_CNT. This bit clears itself automatically.

RESET **Reset DMA** **Bit 7 (Read/Write)**

In its set state, this bit acts as a hardware reset, putting the L64853 into the following initial state: ERR_PEND, INT_EN, FLUSH, DRAIN, WRITE, EN_DMA, REQ_PEND, EN_CNT, TC, PACK_CNT, and BYTE ADDR are all clear; and RESET is set.

WRITE **Read/Write** **Bit 8 (Read/Write)**

This bit determines the direction of the D channel DMA transfer. When set, the flow is to memory (memory write); when cleared, the flow is from memory (memory read).

EN_DMA **Enable DMA** **Bit 9 (Read/Write)**

When set, this bit allows the L64853 to respond to DMA requests by the D channel controller.

REQ_PEND **Request Pending** **Bit 10 (Read Only)**

This bit is set when the D channel controller requests a DMA transfer and remains set as long as D channel DMA transfers are taking place. Do not set RESET or FLUSH while this bit is set. The driver should poll this bit until it is cleared and then take any necessary clean-up action by setting the FLUSH or DRAIN bits.

BYTE ADDR **Byte Address** **Bits 12-11 (Read)**

This two-bit counter contains the lowest two bits of the address of the next byte to be accessed by the D channel controller. Note that the contents of this counter may be different from the address in the DMA Address Counter Register, which contains the address of the next SBus access.

EN_CNT **Enable Counter** **Bit 13 (Read/Write)**

The set state enables operation of the internal Byte Counter Register. The set state also enables operation of the TC bit.

TC **Terminal Count** **Bit 14 (Read Only)**

The set state of this bit reveals that the byte counter has expired. That is, TC is set when a byte count makes a transition from 000001 to 000000. TC is cleared only by the SBUS RESET signal or by setting FLUSH.

ILACC	**ILACC 79C900**	**Bit 15 (Read/Write)**

This bit should be cleared by software. It is present for historical reasons.

Reserved		**Bits 27-16 (Read Only)**

These bits are reserved for future enhancements; they are always clear.

DEV ID	**Device ID**	**Bits 31-28 (Read Only)**

Future revisions of the L64853 may be specified using these bits. For the current implementation, the field is set to 1000 (that is, 8).

DMA Address Counter

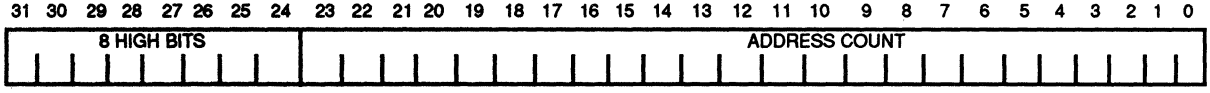


Figure 2.2 DMA Address Counter (DMA_ADDR)

The 32-bit Address Counter (Read/Write; shown in Figure 2.2 above) is used to contain the starting virtual address of a D channel DMA transfer to or from SBus memory. The lower three bytes are an auto-incrementing counter: a 24-bit counter that increments by 1, 2, or 4 during write transfers (to memory), depending on the size of the data item written. During DMA transfers (from memory), the counter always increments by 4.

The upper byte is an 8-bit register that contains the high-order byte of the virtual address. It does not auto-increment; therefore, DMA transfers are limited to 16 MB and may not cross a 16 MB address boundary.

After a RESET, the Address Counter contains an indeterminate value.

DMA Byte Counter

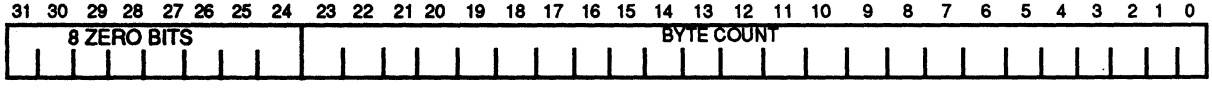


Figure 2.3 DMA Byte Counter (DMA_BCNT)

The 24-bit Byte Counter (Read/Write; shown in Figure 2.3) is used to hold the number of bytes remaining in the current D channel DMA transfer. The counter decrements to 0, at which point it generates an interrupt if INT_EN is set in the CSR. This register is activated by setting EN_CNT in the CSR. The value in this register after RESET is indeterminate.

Even though this is a 24-bit register, it is read as a full word (32-bit) value. The upper byte always reads as zero.

Internal ID Register

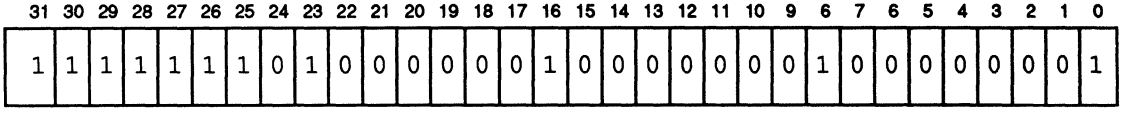


Figure 2.4 ID Register

Each SBus device can be identified by software through a unique ID number supplied by the user of that device. The ID for the L64853 is hardwired to the 32-bit value: FE810101 (see Figure 2.4). Refer to *The SBus Specification* for details on using the External ID Register.

2.3 Addressing Internal and External Registers

Any SBus peripheral device that is built using the L64853 will contain two types of configuration and control registers: the internal L64853 registers described in Section 2.2 and the external registers in the peripheral controllers that are attached to the L64853's two channels.

The system addresses that specify the internal or external registers are determined by the implementation of the SBus Controller and the way that the hardware designer connects the physical address signals.

The SBus uses a geographical addressing scheme that assigns ranges of address to each SBus connector. When the SBus controller detects access to a particular range, it asserts the $\overline{\text{SEL}}$ signal dedicated to the related connector.

When the L64853's $\overline{\text{SEL}}$ input is asserted by the SBus Controller, the L64853 uses four address inputs to further decode the address present on SBus signals PA[27:0]. These four L64853 inputs are PA[X:Y] and PA [3:2]. The PA[X:Y] signals are connected to two of the SBus's 28 physical address lines, PA[27:0]. (On the SPARCstation 1, these two lines are PA[23:22].) The L64853 uses PA[X:Y] to select which category of registers is being accessed according to Table 2.1. The size of the data item that must be transferred over the SBus (specified by SIZ[2:0]) is also shown for each category of register.

PA[X:Y]	Addressed Register Type	Size
00	Internal ID Register	Word
01	Internal programmable registers	Word
10	D channel registers (external)	Byte
11	E channel registers (external)	Half

Table 2.1 Accesses by Register Type

Two other physical address lines PA[3:2] are normally connected to the PA[3:2] pins of the L64853, and, when the PA[X:Y] value is 01, these lines are used to select a particular internal programmable register (see Table 2.1).

PA[3:2]	Addressed Register
00	DMA Control/Status Register
01	DMA Address Register
10	DMA Byte Count Register
11	Reserved for Testing

Table 2.2 Internal Register Addresses

Other physical address lines are wired directly to the two peripheral controllers and are used in an implementation-dependent manner to select specific external registers on these two chips.

Note: PA1 is used by the L64853 to steer the appropriate (upper or lower) half-word of data to the E channel peripheral during slave accesses.

Chapter 3: Operation of the D and E Channels

This chapter describes the functioning of the D and E channels. First, the two SBus operational modes are introduced. Next, the two basic kinds of operations (DMA Reads and Writes and Register Reads and Writes) are described in step-by-step fashion. Finally, details pertinent to the individual channels are provided. This chapter is directed particularly to system programmers.

3.1 Modes of Operation

The L64853 supports two basic modes of operation:

- Master mode

  In Master mode, the L64853 is an SBus DVMA Master and performs DMA reads and writes. The L64853 generates virtual addresses on the SBus data lines, which are translated into physical addresses by the SBus controller's MMU, and transfers data between memory and a peripheral controller.

- Slave mode

  In Slave mode, the CPU is the SBus master, and it performs register reads and writes of the L64853 as an SBus slave. The physical address lines of the SBus access the internal registers of the L64853 and those of the two peripheral controllers, using a register addressing scheme that is implementation dependent.

The basic operations of both the E and D channels are DMA Reads and Writes – executed when the L64853 is in Master mode – and Register Reads and Writes, executed when the L64853 is an SBus Slave.

The main difference between the two channels is that the E channel is programmed entirely through the external registers – none of the internal L64853 registers affect the E channel.

DMA transfers are accomplished by the coordinated activity of all chips in the system: the peripheral controller, the L64853, the CPU, the SBus controller, and the memory controller. First the procedure for a write operation is presented, then differences required for a read are noted.

3.2 DMA Reads and Writes (Master Mode)

The following are the steps of the write operation:

1. One of the two peripheral controllers requests a DMA transfer, using either the Hold signal ($\overline{\text{E\_HOLD}}$ in the E channel) or Request signal (D_REQ in the D channel).

2. The L64853 acknowledges the requesting peripheral controller, using either the Hold Acknowledge signal ($\overline{\text{E\_HLDA}}$ in the E channel) or Acknowledge signal ($\overline{\text{D\_ACK}}$ in the D channel), thus prompting the controller to begin the data write operation.

3. (E channel only) The L64853 receives a 24-bit address from the E channel controller on the E_AD[15:0] and E_A[23:16] lines into the Address Latch Register. This address serves as the lowest 24 bits of the memory address to which the data is

written. This address was set up by an earlier slave-mode transfer between the CPU and the E channel controller, via the L64853.

4. The L64853 loads 8-bit or 16-bit data items from the peripheral controller into the appropriate Pack/Unpack Register until the register is full, using the E_AD[15:0] lines (E channel) or D_D[7:0] lines (D channel).

5. The L64853 requests control of the SBus using the Bus Request signal ($\overline{BR}$); control is granted by the SBus Controller with the Bus Grant signal ($\overline{BG}$).

6. The L64853 outputs a 32-bit virtual address from either the Address Latch Register (E channel) or the Address Counter Register (D channel) onto the SBus data lines D[31:0]. In an E channel operation, the 32-bit address is formed by appending FF as the high-order byte to the 24-bit address in the Address Latch Register. (The address in the Address Count Register must have been previously set up by an earlier slave mode transfer from the CPU.) If both D and E channels need the SBus simultaneously, the E channel has priority.

7. The SBus Controller's MMU translates this virtual address into a 28-bit physical address on the SBus physical address lines PA[27:0].

8. The L64853 outputs the contents of the appropriate Pack/Unpack Register onto the SBus data lines D[31:0] for transfer to the memory controller, which sends the data to the memory array.

Read operations are very similar:

1. One of the two peripheral controllers requests a DMA transfer, using either the Hold signal ($\overline{E\_HOLD}$ in the E channel) or Request signal (D_REQ in the D channel).

2. (E channel only) The L64853 acknowledges the E channel peripheral using $\overline{E\_HLDA}$, then waits for $\overline{E\_DAS}$ to be asserted. When $\overline{E\_DAS}$ is asserted, the L64853 requests control of the SBus.

   On the D channel, if the Unpack Register is empty (no bytes left over from a previous DMA read operation), the L64853 requests control of the SBus without providing acknowledgment ($\overline{D\_ACK}$) to the D channel peripheral. If the Unpack Register has any valid data remaining, the L64853 asserts ($\overline{D\_ACK}$) and begins transferring the data to the D channel peripheral.

3. When the L64853 gets control of the SBus ($\overline{BG}$ is asserted), the L64853 outputs the address of the data requested, either from the Address Latch Register (E channel) or Address Counter (D channel), onto the SBus data lines.

4. The SBus controller's MMU translates this virtual address into a 28-bit physical address on the SBus physical address lines PA[27:0].

5. The L64853 reads the SBus data lines D[31:0] when the data comes back from memory and transfers their contents to the appropriate Pack/Unpack Register.

6. L64853 sends 8-bit or 16-bit data items to the peripheral controller from the appropriate Unpack Register until the register is empty, using the E_AD[15:0] lines (E channel) or D_D[7:0] lines (D channel).

The above summary omits some signals, in particular the Address Strobes ($\overline{\text{E\_AS}}$ and $\overline{\text{AS}}$), and the Read and Write Strobes (RD, E_READ, $\overline{\text{D\_RD}}$, $\overline{\text{D\_WR}}$). Their operation is covered in Chapter 4 (refer particularly to the timing diagrams). When all the data bytes have been transferred, the L64853 can generate an interrupt that informs the CPU that the operation is complete (this occurs only when the Byte Counter is zero and interrupts are enabled).

3.3 Register Reads and Writes (Slave Mode)

Register reads and writes allow the CPU to read status conditions and to set up data transfer parameters. Register reads and writes are simpler to describe than DMA transfers, because the CPU is the Bus Master throughout, and the L64853 is passive.

The following are the steps for a register read or write operation:

1. The CPU executes a LOAD or STORE instruction using the system-dependent address that causes the Slave Select signal ($\overline{\text{SEL}}$) connected to the L64853 to be asserted.

2. The SBus controller converts the CPU's LOAD or STORE address into an active $\overline{\text{SEL}}$ signal for the L64853 and a pattern on the physical address lines PA[27:0], along with several other active strobes specifying an address and a read or write (see the timing diagram in **Chapter 4: Interface Description.**)

3. The L64853 recognizes that the read or write access is intended for itself (the Slave Select is specific), and interprets the pattern on the physical address lines to determine which register (internal, external E channel controller, or external D channel controller) is the intended target of the read or write. (Refer to **Chapter 2: Registers** for the assignment of physical addresses to L64853 internal registers.) **Chapter 6: Applications** describes the register addresses for the external registers on the SPARCstation 1.

4. If the target register is internal to the L64853 (that is, a 32-bit register), the data is either read from the intended register to the data lines D[31:0] (reads) or written from the data lines to the intended register (writes).

5. If the target register is external, the data transfer over the SBus must be either byte (D channel) or half word (E channel). The L64853 passes the data between the SBus and the target controller without any buffering in the relevant Pack/Unpack Register. For the E channel, the data is transferred over the E_AD[15:0] lines to or from a 16-bit register in the E channel controller; for the D channel, the data is transferred over the D_D[7:0] lines to or from an 8-bit register in the D channel controller. To accomplish this transfer, the L64853 also asserts various strobes in the D or E channels (refer to **Chapter 4: Interface Description** for timing diagrams).

6. Finally, the L64853 asserts the SBus Acknowledge signal ($\overline{ACK2}$ in almost all cases), thus informing the SBus controller that the slave-mode operation has completed.

7. If the D or E channel is busy transferring data to or from the Pack/Unpack Register when the CPU selects that channel, the DMA chip will assert Rerun Acknowledgment.

3.4 Specifics of D Channel Operation

For write operations (to memory) four 8-bit D channel transfers are converted into one 32-bit SBus cycle, whenever possible, through use of the Pack Register. However, initial bytes in the transfer that are not word-aligned ("leading fragments") are transferred to memory as bytes until the address reaches a word boundary; then the SBus transfers are by words. Final bytes that do not fill a word ("trailing fragments") must be explicitly written to memory by setting the DRAIN bit in the DMA Control/Status Register.

The 2-bit counter (PACK_CNT) in the DMA Control/Status Register tracks the number of bytes stored in the internal Pack Register during DMA read or write transfers. During writes, whenever the PACK_CNT = 3 and another byte is accepted, a word write is scheduled with the SBus interface. If a DMA transfer completes, leaving a non-word fragment in the Pack Register, this counter is used by the hardware to determined how many bytes to write to the SBus when DRAIN is set.

The DMA Address Register, which contains the memory address of the DMA transfer, always points at the next memory location to be accessed. When the direction of a DMA transfer is to memory (writes), the counter is incremented by the size of the write (1 byte or 4 bytes) upon completion of the transfer.

When the direction of the DMA transfer is from memory (reads), the address is always incremented by 4 (32 bits being received simultaneously). All DMA reads are thus word-length, and all are word-aligned (that is, the lowest two bits of the address are always zero).

Read DMA operations always transfer 32 bits into the Unpack Register, from which four consecutive bytes are unpacked and written to the D channel controller. Every time the byte address crosses a word boundary, a new word is read from memory, the address is incremented by 4, and the PACK_CNT field in the DMA Control/Status Register is set to 3 (that is, the first byte is sent to the peripheral controller immediately). The next three transfers to the D channel controller use data from the Unpack Register. Each transfer causes PACK_CNT to be decremented. Setting FLUSH in the DMA Control/Status Register or writing a new address in the DMA Address Register causes the L64853 to mark any data left in the Unpack Register during a read as invalid and forces a new memory access.

Since the DMA Address Register is incremented by the SBus interface of the L64853 when it makes SBus transfers, this address may occasionally disagree with the value in the BYTE_ADDR field of the DMA Control/Status Register. (This 2-bit byte counter always contains the lowest two bits of the next byte location that the D channel controller will access.) BYTE_ADDR increments by 1 each time a byte is transferred between the peripheral controller and the L64853. BYTE_ADDR is loaded at the same time an address is loaded into the DMA Address Register, and it receives the two least-significant bits of the address.

Termination and Interrupts

Three control bits in the DMA Control/Status Register are used by the device driver to terminate a transfer: EN_DMA, DRAIN, and FLUSH. Clearing the EN_DMA bit causes new DMA requests from the D channel controller to be ignored. The current DMA transfer, however, is unaffected by this bit. EN_DMA can be set or cleared at any time without affecting a transfer currently in progress.

DRAIN sends all data currently in the Unpack Register to memory and clears PACK_CNT. It is useful when a transfer completes and the data for transfer to memory does not fill the 32-bit word. DRAIN can also be used to leave a transfer that has been stopped with EN_DMA in a clean state so that it can be restarted later. Because a DRAIN sequence leaves the DMA Address Register pointing to the byte address beyond the last byte or word written, the address counter must be reloaded prior to the next transfer so that the byte counter, BYTE_ADDR, is set correctly.

FLUSH provides a mechanism for the driver to clean up the state of a transfer (clear PACK_CNT) without draining the packed data to memory. It is also used to clear the ERR_PEND indicator, allowing error conditions to be cleared cleanly.

The RESET bit of the DMA Control/Status Register generates an external reset signal that resets all D channel interface logic (state machines). If any memory activity is still pending, as indicated on the REQ_PEND bit, do *not* set RESET and/or FLUSH. If REQ_PEND is asserted, the driver should poll it until it is de-asserted. Similarly, writing to the DMA Address Register, changing the WRITE bit in the DMA Control/Status Register, or writing the BYTE_CNT Register when the REQ_PEND bit is set puts the L64853 into an unknown state.

The INT_EN bit is provided to enable or disable the generation of an interrupt on the SBus, whether these interrupts arise from interrupts from the D channel controller, or from errors during memory accesses, or completion of a transfer. Interrupts from the D channel controller are visible as INT_PEND in the DMA Control/Status Register. An error condition that occurs during a memory access (for example, a parity error, timeout, or protection error) causes the ERR_PEND bit to be set. Similarly, expiration of the BYTE_CNT when enabled causes the INT_PEND bit to be set. Any of these conditions can cause the L64853 to generate an SBus interrupt, if interrupts are enabled by INT_EN.

Use of the Byte Counter Register

The Byte Counter Register can be used with some D channel controllers. To work with this Register, perform the following operations:

1. Load the DMA virtual address into a DMA Address Register.
2. Load the byte count into a DMA Byte Counter.
3. Load the peripheral device with relevant commands.
4. Load the DMA Control/Status Register with enables and direction bits (EN_DMA, EN_CNT, WRITE).

Data is transferred until the byte count expires at which point the TC Flag in the DMA Control/Status Register is set and an interrupt is generated, if enabled by INT_EN.

5. After the CPU has serviced the interrupt and is ready to start another DMA operation, set FLUSH and begin again with Step 1.

3.5 Specifics of E Channel Operation

As mentioned earlier, the E channel, like the D channel, supports DMA read and write operations, as well as read and write operations to the external Registers of the channel controller. The E channel, however, is programmed through the external Registers. The only Registers on the L64853 dedicated to the E channel are the Address Latch and the E channel Pack/Unpack Registers.

During a DMA read or write operation, the E channel controller provides the memory address to which the DMA data is to be written. This 24-bit address from the controller is transferred over the E_A[23:16] and E_AD[15:0] lines to the Address Latch on the L64853. The L64853 then appends a high-order 8-bit value FF to this 24-bit address to create the 32-bit SBus virtual address, which it sends out on the SBus data lines D[31:0] during a DVMA Master cycle.

The data on the SBus during the data transfer portion of the DVMA Master cycle is transferred between memory and the E channel Unpack Register. DMA transfers from memory (reads) always transfer 32 bits into the Unpack Register, from which two consecutive half-words are unpacked and written to the E channel controller. Every time the address crosses a word boundary a new word is read from memory. The next transfer to the E channel controller uses data from the Unpack Register.

When the E channel controller is transferring data to memory, the data is packed, unless the access is either not word-aligned, and there is no data buffered, or the data involved is a single byte, in which case the L64853 transfers the data to memory immediately. Data that is word-aligned is held pending the next sequential write, assuming it is a part of the same E channel controller burst transfer, whereupon the entire word is transferred to memory.

Any data in the Pack Register is transferred to memory whenever $\overline{\text{E\_HOLD}}$ is deasserted. An example of such an event would be a single half-word message descriptor write by the E channel controller.

The E channel supports burst mode transfers (that is, multiple half-word transfers before the $\overline{\text{E\_HOLD}}$ signal is released). At the end of read bursts, data left in the Pack/Unpack Register is discarded. At the end of write bursts, data remaining in the Pack Register is automatically written to memory.

Chapter 4: Interface Description

This chapter describes the signals used by the L64853 and the data transfer cycles for the D and E channels. This information is useful primarily for hardware designers.

4.1 Pin Summary

Figure 4.1 shows the pin name abbreviations associated with each pin of the L64853. Table 4.1 provides the signal name, abbreviation, and pin number.

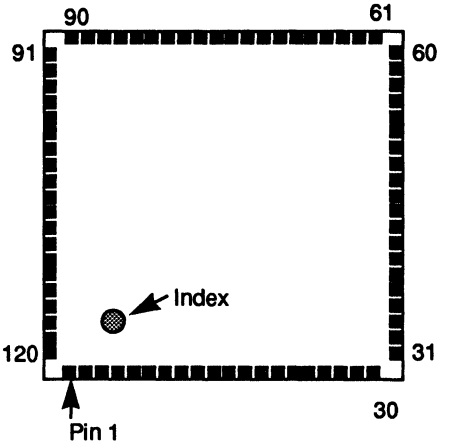


Figure 4.1 L64853 Pin Diagram (Top View)

Pin	Abbreviation	Pin	Abbreviation	Pin	Abbreviation	Pin	Abbreviation
1	$\overline{\text{SLOW}}$	31	D_D7	61	VDD	91	VSS
2	$\overline{\text{TEST}}$	32	D_D6	62	E_AD1	92	$\overline{\text{INTREQ}}$
3	OD_TST	33	D_D5	63	E_AD0	93	D8
4	PA3	34	D_D4	64	E_AD13	94	D9
5	PA2	35	D_D3	65	E_AD14	95	D10
6	PA1	36	D_D2	66	E_AD15	96	D11
7	VDD	37	D_D1	67	E_AH16	97	D12
8	VSS	38	D_D0	68	E_AH17	98	D13
9	PAX	39	D_RESET	69	E_AH18	99	D14
10	PAY	40	$\overline{\text{D\_IRQ}}$	70	E_AH19	100	D15
11	$\overline{\text{LERR}}$	41	$\overline{\text{D\_CS}}$	71	VSS	101	D16
12	$\overline{\text{BG}}$	42	VSS	72	VDD	102	D17)
13	$\overline{\text{BR}}$	43	VDD	73	E_AH20	103	unconnected
14	$\overline{\text{ACK0}}$	44	CLK	74	E_AH21	104	VDD
15	$\overline{\text{ACK2}}$	45	$\overline{\text{D\_RD}}$	75	E_AH22	105	VSS
16	SIZ2	46	$\overline{\text{D\_WR}}$	76	E_AH23	106	D18
17	SIZ1	47	D_REQ	77	$\overline{\text{ACK1}}$	107	D19
18	SIZ0	48	$\overline{\text{D\_ACK}}$	78	$\overline{\text{RESET}}$	108	D20
19	VDD	49	E_AD12	79	RD	109	D21
20	unconnected	50	E_AD11	80	VSS	110	D22
21	$\overline{\text{ID\_CS}}$	51	E_AD10	81	$\overline{\text{SEL}}$	111	D23
22	E_BYTE	52	VSS	82	$\overline{\text{AS}}$	112	D24
23	$\overline{\text{E\_CS}}$	53	E_AD9	83	D0	113	D25
24	E_READ	54	E_AD8	84	D1	114	VSS
25	$\overline{\text{E\_HLDA}}$	55	E_AD7	85	D2	115	D26
26	$\overline{\text{E\_HOLD}}$	56	E_AD6	86	D3	116	D27
27	$\overline{\text{E\_AS}}$	57	E_AD5	87	D4	117	D28
28	$\overline{\text{E\_DAS}}$	58	E_AD4	88	D5	118	D29
29	$\overline{\text{E\_RDY}}$	59	E_AD3	89	D6	119	D30
30	VSS	60	E_AD2	90	D7	120	D31

Table 4.1 Signal Summary

4.2 Signal Descriptions

For ease of reference, the signal descriptions are divided into four functional groups: D channel, E channel, SBus interface, and miscellaneous. Within each group, the signals are listed alphabetically by abbreviation. The SBus interface descriptions are minimal; the reader is referred to *The SBus Specification* for additional information.

D Channel Signals

These D channel signals define an interface to an external 8-bit controller chip:

D_DACK **DMA Acknowledge.** (*Output buffer, active low, 3-state, CMOS, 4 mA output drive, Pin 48.*)

DMA Acknowledge is used to notify the 8-bit controller when it has been granted a DMA read or write cycle; that is, when it is time for the 8-bit controller to transfer data through the L64853 to or from the SBus. DMA Acknowledge and DMA Chip Select must never be active at the same time.

D_CS **DMA Chip Select.** (*3-state output buffer, active low, CMOS, 4 mA output drive, Pin 41.*)

DMA Chip Select is used to select the 8-bit controller as an I/O device; that is, to read or write the 8-bit controller's internal registers. DMA Chip Select and DMA Acknowledge must never be active at the same time.

D_D[7:0] **DMA Data Bus.** (*Bidirectional buffer, active high, TTL input levels, internal pulldown, 4 mA output drive, Pins 31-38.*)

Data lines D_D[7:0] connect the 8-bit controller and the L64853. They send data to and from the D channel Pack/Unpack Registers in the L64853 or send data through internal L64853 logic where it emerges as the lower 8 bits of the 32-bit L64853 data bus. These signals also serve as the input for external ID data when $\overline{\text{ID\_CS}}$ is asserted ($\overline{\text{D\_RD}}$ must also be asserted).

$\overline{\text{D\_IRQ}}$ **DMA Interrupt Request.** (*Input buffer, active low, TTL level, non-inverting, internal pullup, Pin 40.*)

DMA Interrupt Request is pulled low when the 8-bit controller signals an interrupt; for instance, when a data transfer is completed and the 8-bit controller signals the completion to the L64853.

$\overline{\text{D\_RD}}$ **DMA Read Strobe.** (*3-state output buffer, active low, CMOS, 4 mA output drive, Pin 45.*)

The L64853 uses DMA Read Strobe to signal a read access to the 8-bit controller. This signal is used both for reading 8-bit controller internal registers and during actual data transfers from the D channel controller through the L64853 to the SBus (that is, during a memory write). DMA Chip Select or DMA Acknowledge must also be active.

D_REQ **DMA Request.** *(Input buffer, active high, TTL level, non-inverting, Pin 47.)*

DMA Request is used by the D channel controller to request that it be allowed to perform a DMA operation.

D_RESET **DMA Reset.** *(3-state output buffer, active high, CMOS, 4 mA output drive, Pin 39.)*

DMA Reset is an output that can be used to clear the 8-bit controller.

$\overline{\text{D\_WR}}$ **DMA Write Strobe.** *(3-state output buffer, active low, CMOS, 4 mA output drive, Pin 46.)*

The L64853 uses DMA Write Strobe to signal a write access to the 8-bit controller. This signal is used both for writing 8-bit controller internal registers and during actual data transfers from the D channel controller through the L64853 to the SBus (that is, during a memory read). DMA Chip Select or DMA Acknowledge must also be active.

$\overline{\text{ID\_CS}}$ **Secondary Device Select.** *(Bidirectional buffer, active low, TTL input levels, 4 mA output drive, Pin 21.)*

The Secondary Device Select line is pulled high to specify the existence of an external PROM. The DMA Data Bus lines are the input for $\overline{\text{ID\_CS}}$ ($\overline{\text{D\_RD}}$ must also be asserted). For more details about such a PROM, see *The SBus Specification.*

$\overline{\text{SLOW}}$ **Fast/Slow DMA Acknowledge Cycles.** *(Input buffer, active low, TTL level, non-inverting, internal pullup, Pin 1.)*

The $\overline{\text{SLOW}}$ signal is used to specify either fast or slow mode. The pin controls transfer speed on the D channel. Fast mode must be used with the ESP SCSI controller. This signal requires no external connection. When $\overline{\text{SLOW}}$ equals ground (GND), two delay cycles are added.

E Channel Signals

These E channel signals define an interface to an external 16-bit E channel controller chip:

E_AD[15:0] **E Channel Address/Data Bus.** *(Bidirectional buffer, active high, TTL input levels, internal pullup, 4 mA output drive, Pins 49-51, 53-60, 62-66.)*

E_AD[15:0] are address/data bus lines connecting the E channel controller and the L64853. These lines send data to and receive data from the E channel Pack/Unpack Registers in the L64853 or send data through internal L64853 logic, where it emerges as the lower half word of the 32-bit L64853 data bus. These lines also send the lower 16 bits of virtual addresses to the address latch in the L64853. For register reads/writes, the E channel controller is the driver.

E_A[23:16] **E Channel Address.** (*Input buffer, active high, TTL level, non-inverting, internal pulldown, Pins 67-70, 73-76.*)

Lines E_A[23:16], which contain the upper 8 bits, together with the E Channel Address/Data Bus 15-0 lines, contain the 24-bits of a virtual address.

$\overline{\text{E\_AS}}$ **E Channel Address Strobe.** (*Input buffer, active low, TTL level, non-inverting, internal pulldown, Pin 27.*)

E Channel Address Strobe can be driven low by the E channel controller to signal that an address transfer is taking place.

E_BYTE **E Channel Byte Marker.** (*Input buffer, active high, TTL level, non-inverting, internal pullup, Pin 22.*)

When high, E Channel Byte Marker specifies that a byte transfer is taking place on the data lines (E Channel Address/Data 15:0); when clear, it indicates that a half-word transfer is taking place. The lowest bit of the address determines which byte is effective, if (E Channel Byte Marker) is set:

E_BYTE	E_AD(0)	
high	low	lower byte
high	high	upper byte

$\overline{\text{E\_CS}}$ **E Channel Chip Select.** (*3-state output buffer, active low, CMOS, 4 mA output drive, Pin 23.*)

E Channel Chip Select is used to select the E channel controller as an I/O device; that is, to read or write the E channel controller's internal registers.

$\overline{\text{E\_DAS}}$ **E Channel Data Strobe.** (*Bidirectional buffer, active low, TTL input levels, internal pullup, 4 mA output drive, Pin 28.*)

E Channel Data Strobe is high during the address portion of a memory access and low during the data portion. This signal is used to specify that a data transfer is taking place. This signal is driven by the L64853 for slave accesses.

$\overline{\text{E\_HLDA}}$ **E Channel Hold Acknowledge.** (*3-state output buffer, active low, CMOS, 4 mA output drive, Pin 25.*)

The L64853 uses E Channel Hold Acknowledge to acknowledge a prior E Channel Hold signal from the E channel controller, and thus to signal the controller to proceed with a DMA transfer.

$\overline{\text{E\_HOLD}}$ **E Channel Hold.** (*Input buffer, active low, TTL level, non-inverting, internal pullup, Pin 26.*)

E Channel Hold is asserted by the E channel controller when it requires access to memory. E Channel Hold is held low for the entire ensuing bus

transaction. Once E Channel Hold goes inactive (high), so does E Channel Hold Acknowledge.

$\overline{\text{E\_RDY}}$ **E Channel Ready.** *(Bidirectional buffer, TTL input levels, internal pullup, 4 mA output drive, Pin 29.)*

The E Channel Ready strobe can be used in conjunction with the data lines to signal a data transfer.

E_READ **E Channel Read.** *(Bidirectional buffer, active high, TTL input levels, internal pullup, 4 mA output drive, Pin 24.)*

E Channel Read indicates the type of operation to be performed in the current bus cycle. When high, E Channel Read signals a read from the memory to the controller. When low, it signals a write from the controller to the memory. This signal is driven by the L64853 during slave accesses.

SBus Interface Signals

These SBus signals interface the L64853 to the other system components:

PA[X:Y] **SBus Physical Address X:Y.** *(input buffer, active high, TTL level, non-inverting, internal pullup, Pins 9-10.)*

SBus Physical Address X and SBus Physical Address Y are two address lines (x and y) used for slave cycle address decodes.

PA[3:1] **SBus Physical Address.** *(input buffer, active high, TTL level, non-inverting, internal pullup, Pins 4-6.)*

Three lines, SBus Physical Address, are used to carry physical address signals.

$\overline{\text{ACK1}}$ **SBus Acknowledge 1.** *(Bidirectional buffer, active low, TTL input levels, inverting, internal pullup, 4 mA output drive, Pin 77.)*

SBus Acknowledge 1 is used to acknowledge an 8-bit slave cycle.

$\overline{\text{ACK2}}$ **SBus Acknowledge 2.** *(Bidirectional buffer, active low, TTL input levels, inverting, internal pullup, 4 mA output drive, Pin 15.)*

SBus Acknowledge 2 is used to acknowledge a 32-bit slave cycle.

$\overline{\text{AS}}$ **SBus Address Strobe.** *(Input buffer, active low, TTL level, non-inverting, internal pullup, Pin 82.)*

SBus Address Strobe low signals the address is valid.

$\overline{BG}$ **SBus Bus Grant.** *(Input buffer, active low, TTL level, non-inverting, internal pullup, Pin 12.)*

A low SBus Bus Grant signals that the CPU has granted control of the SBus to the L64853.

$\overline{BR}$ **SBus Bus Request.** *(3-state output buffer, active low, CMOS, 4 mA output drive, Pin 13.)*

To perform a master cycle, the L64853 first drives SBus Bus Request low to request control of the SBus bus from the CPU.

CLK **SBus Clock.** *(input clock driver, Pin 44.)*

The SBus Clock signal controls internal operations and rates of data transfer. It is usually derived from the master system clock or an associated CPU clock.

D[31:0] **SBus Data Bus.** *(Bidirectional buffer, TTL input levels, internal pullup, 4 mA output drive, Pins 83-90, 93-102, 106-113, 115-120.)*

Lines D[31:0] are the data lines of the SBus.

$\overline{ACK0}$ **SBus Acknowledge 0.** *(Bidirectional buffer, active low, TTL input levels, inverting, internal pullup, 4 mA output drive, Pin 14.)*

SBus Acknowledge 0 is used to report errors.

$\overline{INTREQ}$ **SBus Interrupt Request.** *(Bidirectional buffer, active low, TTL input levels, open drain output, 4 mA output drive, Pin 92.)*

SBus Interrupt Request is the SBus interrupt line. It is activated when INT_EN is set and an external interrupt or various error conditions on the L64853 occur.

$\overline{LERR}$ **SBus Late Error.** *(Input buffer, active low, TTL level, non-inverting, internal pullup, Pin 11.)*

SBus Late Error is an error line.

RD **SBus Read/Write.** *(Bidirectional buffer, low = write, 3-state, TTL input levels, internal pullup, 4 mA output drive, Pin 79.)*

SBus Read/Write is used to signal the direction of data transfers on the SBus (reads or writes).

$\overline{RESET}$ **SBus Reset.** *(Input buffer, active low, TTL level, non-inverting, internal pullup, Pin 78.)*

SBus Reset is the SBus reset line.

$\overline{SEL}$ **SBus Select.** *(Input buffer, active low, TTL level, non-inverting, internal pullup, Pin 81.)*

SBus Select is the slave select line specific for the L64853.

SIZ[2:0] **SBus Transfer Size.** *(Bidirectional buffer, active high, TTL input levels, internal pullup, 4 mA output drive, Pins 16-18.)*

The SIZ[2:0] lines are used to specify the size of the data transfer on the SBus: 8 bit, 16 bit, or 32 bit.

Miscellaneous Signals

OD_TST **Parametric Test Output.** *(3-state output buffer, active high, CMOS, 4 mA output drive, Pin 3.)*

$\overline{OD\_TST}$ is a test line.

$\overline{TEST}$ **Test Control.** *(Input buffer, active low, CMOS level, non-inverting, internal pullup, Pin 2.)*

When low, Test Control enables parametric test output and disables all 3-state drivers. When high, TEST disables parametric test output.

VDD **Power.** *(Pins 7, 19, 43, 61, 72, 104.)*

There are six +5 volt power lines.

VSS **Ground.** *(Pins 8, 30, 42, 52, 71, 80, 91, 105, 114.)*

There are nine ground pins.

Pins 20 and 103 are not connected.

4.3 Data Transfer Cycles

The following descriptions and timing diagrams refer to relative timing relationships of L64853 signals during basic operations. For exact timing information, refer to the composite timing diagrams in **Chapter 5: Specifications**.

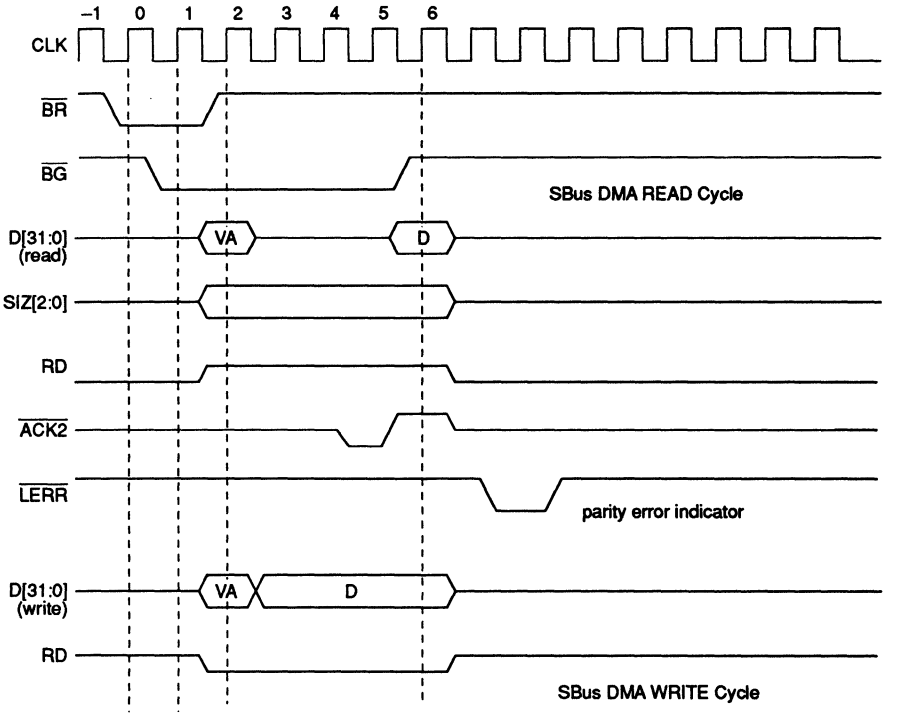
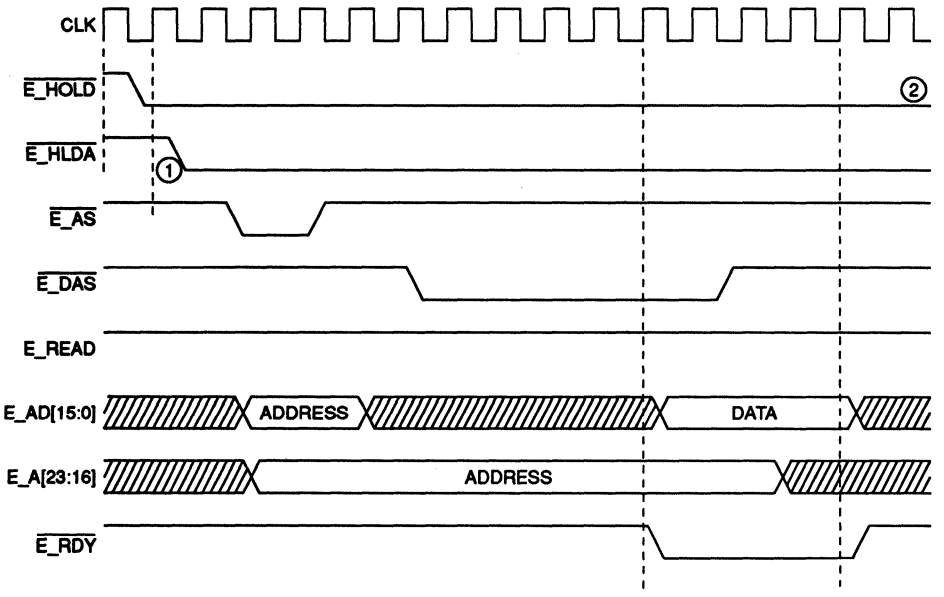
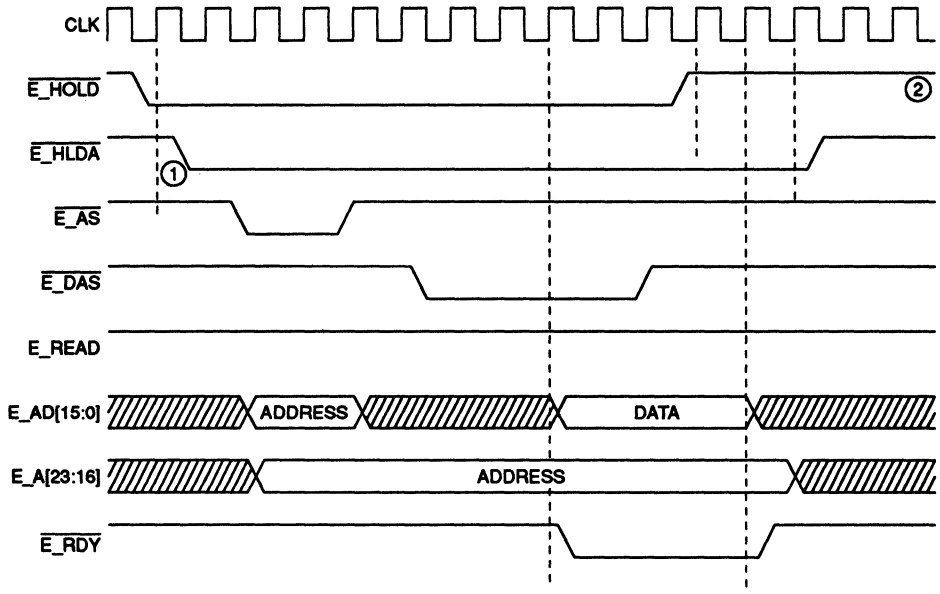


Figure 4.2 SBus DMA Read/Write



1. $\overline{\text{E\_HLDA}}$ is asserted only when the interface is not busy.
2. $\overline{\text{E\_HOLD}}$ stays asserted for burst mode accesses. $\overline{\text{E\_HLDA}}$ must follow it.

Figure 4.3 E Channel DMA Read Cycle (Data not in Pack/Unpack Register)



1. E_HLDA is asserted only when the interface is not busy.
2. E_HOLD stays asserted for burst mode accesses. E_HLDA must follow it.

Figure 4.4 E Channel DMA Read Cycle (Data in Pack/Unpack Register)

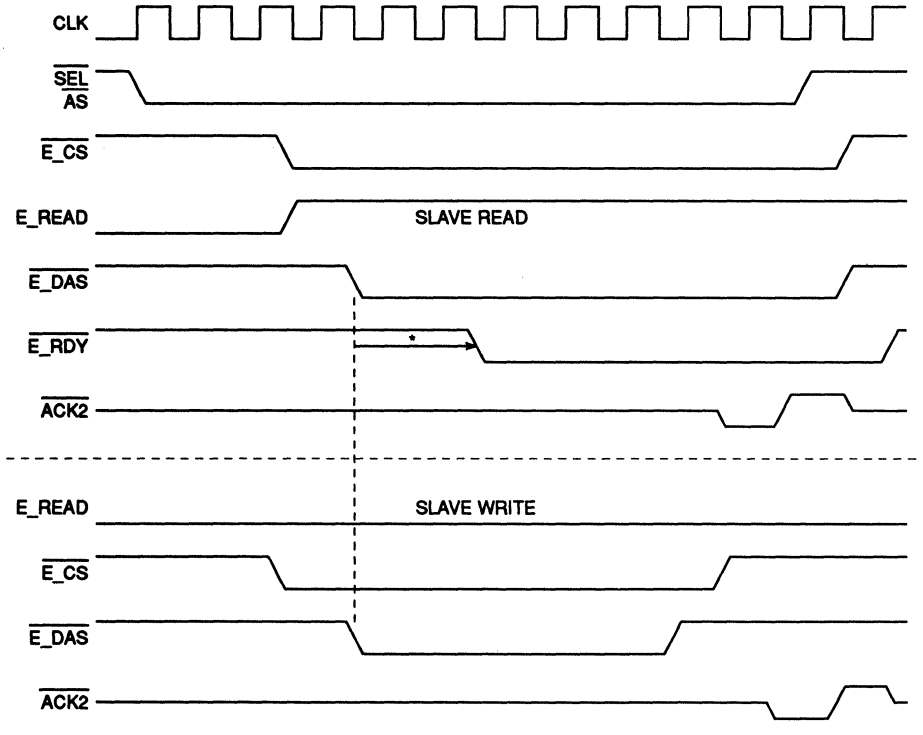
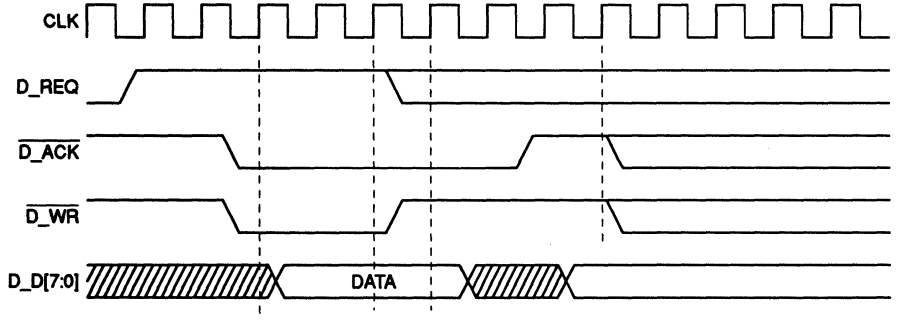
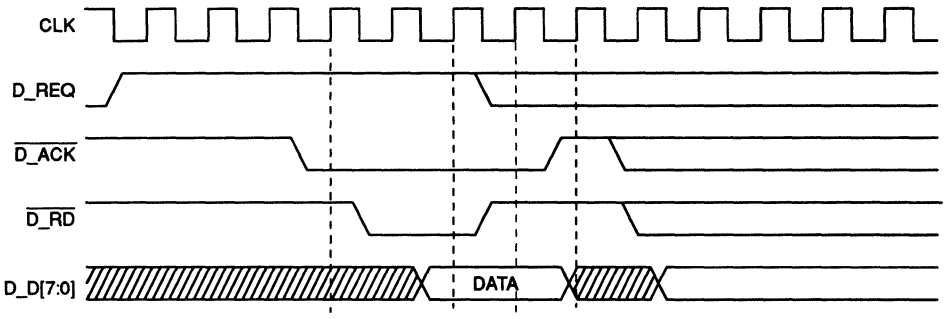


Figure 4.5 E Channel Register Read/Write Cycles



1. A Read indicates a transfer from memory to DMA device.
2. Data is available in Unpack Register. When Unpack Register is empty a memory read must occur to lengthen this operation (see SBus Read Cycle).

Figure 4.6 D Channel DMA Read—Fast Cycle



1. A Write indicates a transfer from DMA device to memory.

Figure 4.7 D Channel DMA Write—Fast Cycle

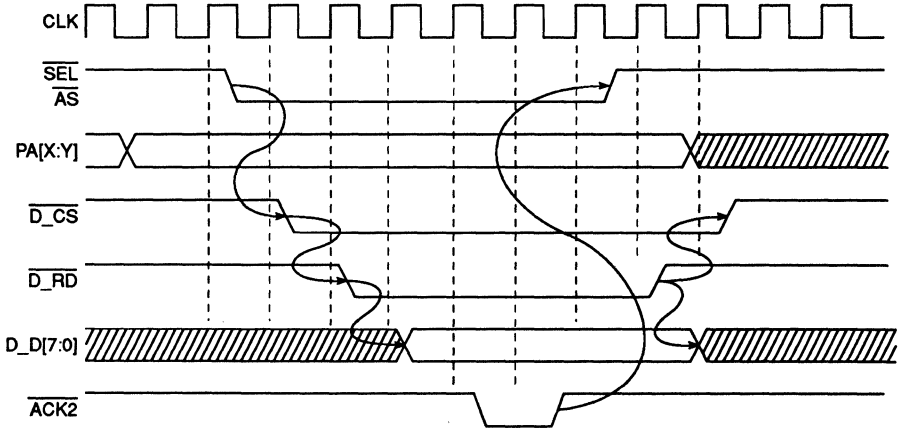


Figure 4.8 D Channel Register Read—Fast Cycle

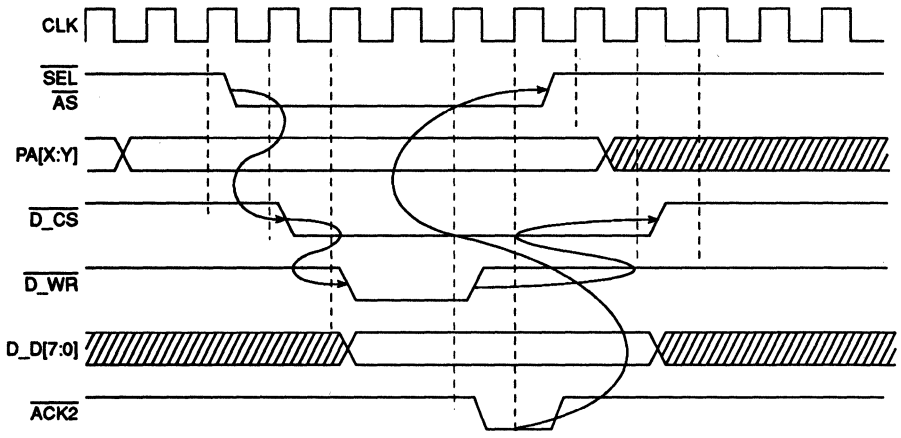
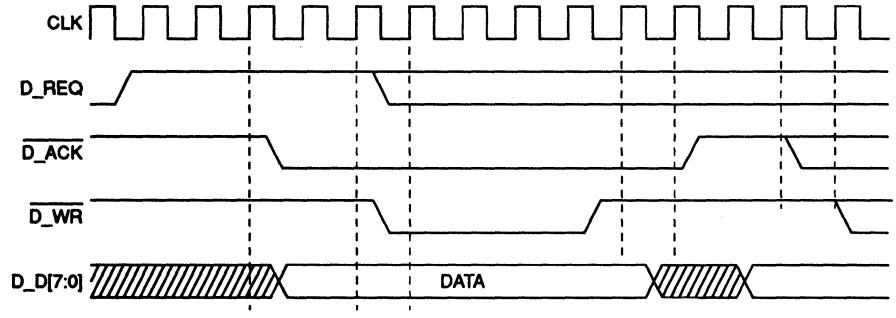


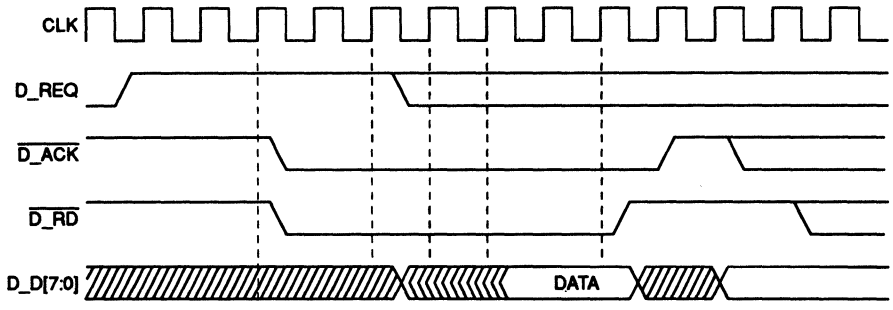
Figure 4.9 D Channel Register Write—Fast Cycle



1. A Read indicates a transfer from memory to DMA device.

2. Data is available in Unpack Register.

Figure 4.10 Extended D Channel DMA Read—Slow Cycle



1. A Write indicates a transfer from DMA device to memory.

Figure 4.11 Extended D Channel DMA Write—Slow Cycle

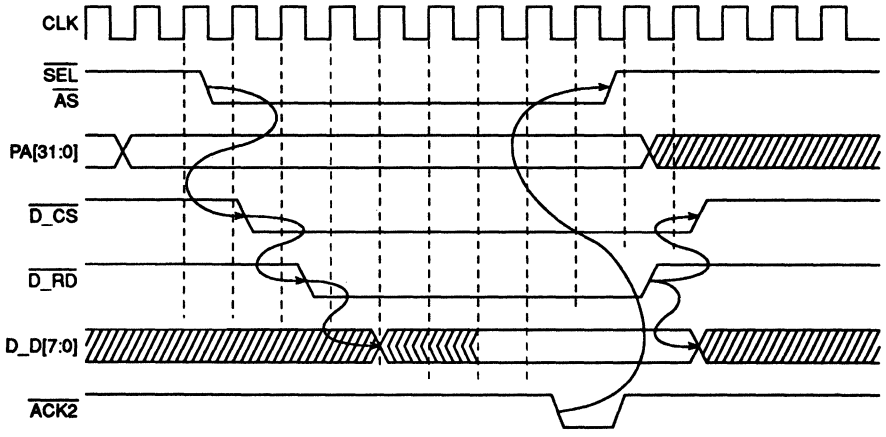


Figure 4.12 Extended D Channel DMA Register Read—Slow Cycle

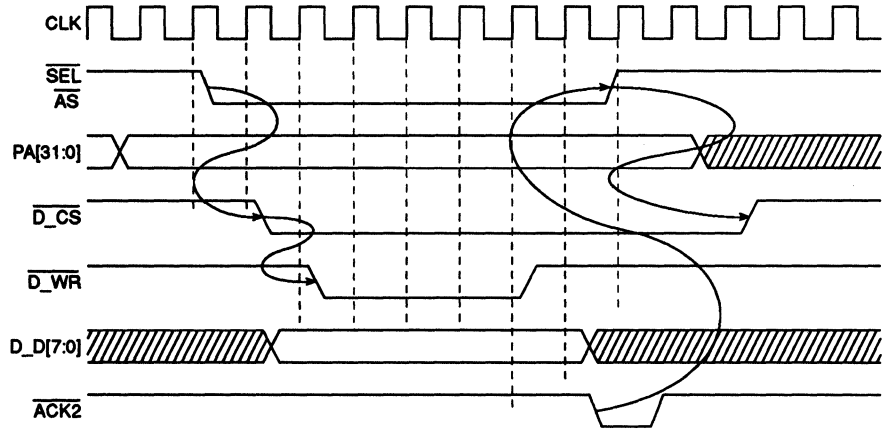


Figure 4.13 Extended D Channel DMA Register Write—Slow Cycle

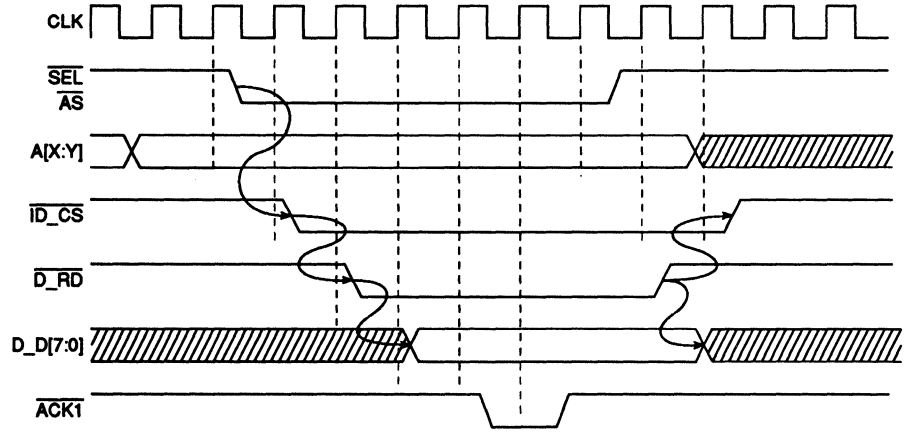


Figure 4.14 Offboard ID Read—Fast Cycle

Chapter 5: Specifications

This chapter provides the AC, DC, environmental, and mechanical specifications of the L64853. This chapter is of use to hardware designers.

5.1 AC Characteristics

The AC characteristics described in the following section apply for V_{DD} = 5V, and TA = 25. Values are in nanoseconds unless otherwise stated.

		(Propagation Delays)			
Symbol	**Input**	**15 pf**	**50 pf**	**85 pf**	**100 pf**
$\overline{BR}$, $\overline{E\_HLDA}$, $\overline{E\_CS}$, $\overline{D\_ACK}$, $\overline{D\_RD}$, $\overline{D\_WR}$, $\overline{D\_CS}$, $\overline{D\_RESET}$	tPHL	2.6	4.1	5.6	6.2
	tPLH	2.4	4.6	6.9	7.8
D[31:0], $\overline{ACK0}$, RD, $\overline{INTREQ}$, SIZ[2:0], E_READ, $\overline{E\_DAS}$, $\overline{E\_RDY}$, E_AD[15:0], D_D[7:0], $\overline{ACK2}$	tPHL	2.7	4.1	5.6	6.3
	tPLH	2.4	4.7	6.9	7.8
$\overline{BG}$, $\overline{RESET}$, $\overline{LERR}$, $\overline{SEL}$, $\overline{AS}$, PA[X:Y], PA[3:1], $\overline{E\_AS}$, $\overline{E\_HOLD}$, $\overline{E\_BYTE}$, E_A[23:16], $\overline{D\_REQ}$, $\overline{D\_IRQ}$		Standard Loads				
		1	2	3	4	5
	tPHL	0.7	0.7	0.7	0.7	8
	tPLH	0.8	0.8	0.9	0.9	1.1

Table 5.1 Signal Propagation Delays

Switching Characteristics

No.	Signal	Description	Conditions	Min.	Max.	Units
1	CLK	clock period		30.0		ns
2		clock high				ns
3		clock low				ns
4	Note 1	hold wrt CLK^		0.0		ns
5	Note 1	setup to CLK^		14.0		ns
6	Note 1	setup to CLK^		23.0		ns
7	Note 1	hold wrt CLK^		5.0		ns
8	Note 1	setup to CLK^		13.5		ns
9	Note 1	hold wrt CLK^		0.0		ns
10	Note 1	CLK^ to output valid	load= 100pf		30.4	ns
11	Note 1	CLK^ to output invalid	load= 100pf		22.0	ns
12	Note 1	CLK^ to output valid	load= 130pf		31.4	ns
13	Note 1	CLK^ to output invalid	load= 130pf		19.7	ns
14	Note 1	CLK^ to output low	load= 100pf		24.0	ns
15	Note 1	CLK^ to output high	load= 100pf		18.5	ns
16	D_REQ	setup to CLK^		0.0		ns
17	D_REQ	hold wrt CLK^		3.7		ns
18	D_D[7:0]	setup to CLK^		2.0		ns
19	D_D[7:0]	hold wrt CLK^		3.5		ns
20	$\overline{\text{D\_IRQ}}$	setup to CLK^		0.0		ns
21	$\overline{\text{D\_IRQ}}$	hold wrt CLK^		4.0		ns
22	$\overline{\text{D\_RD}}$	CLK^ to output low	80pf		25.5	ns
23	$\overline{\text{D\_RD}}$	CLK^ to output high	80pf		20.0	ns
24	$\overline{\text{D\_WR}}$	CLK^ to output low	80pf		22.5	ns
25	$\overline{\text{D\_WR}}$	CLK^ to output high	80pf		18.0	ns
26	D_D[7.0]	CLK^ to output valid	80pf		29.0	ns
^ equals clock high						

Table 5.2 L64853 Switching Characteristics (Part 1 of 3)

No.	Signal	Description	Conditions	Min.	Max.	Units
27	D_D[7.0]	CLK^ to output invalid	80pf		21.0	ns
28	Note 2	CLK^ to output low	80pf		24.0	ns
29	Note 2	CLK^ to output high	80pf		19.0	ns
30	Note 3	D_RD to D_D[7:0] valid			50.0	ns
31	Note 3	$\overline{\text{D\_RD}}$ to D_D[7:0] invalid		0.0		ns
32	D_RESET	CLK^ to output low	80pf		20.0	ns
33	D_RESET	CLK^ to output high	80pf		18.0	ns
34	$\overline{\text{D\_ACK}}$	CLK^ to output low	80pf		23.5	ns
35	$\overline{\text{D\_ACK}}$	CLK^ to output high	80pf		17.0	ns
36	E_AD[15:0]	setup to CLK^ (Note 4)			1.0	ns
37	E_AD(15.0)	hold wrt to CLK^ (Note 4)		4.0		ns
38	E_AD(15:0)	CLK^ to output valid	80pf		36.0	ns
39	E_AD(15:0)	CLK^ to output invalid	80pf		25.0	ns
40	$\overline{\text{E\_HLDA}}$	CLK^ to output high	80pf		18.0	ns
41	$\overline{\text{E\_HLDA}}$	CLK^ to output low	80pf		21.5	ns
42	E_READ	CLK^ to output valid	80pf		15.5	ns
43	E_READ	CLK^ to output invalid	80pf		12.0	ns
44	$\overline{\text{E\_DAS}}$	CLK^ to output valid	80pf		23.0	ns
45	$\overline{\text{E\_DAS}}$	CLK^ to output invalid	80pf		18.5	ns
46	$\overline{\text{E\_RDY}}$	CLK^ to output valid	80pf		23.0	ns
47	$\overline{\text{E\_RDY}}$	CLK^ to output invalid	80pf		17.5	ns
48	$\overline{\text{E\_CS}}$	CLK^ to output high	80pf		15.0	ns
49	$\overline{\text{E\_CS}}$	CLK^ to output low	80pf		20.0	ns
50	$\overline{\text{E\_RDY}}$	setup to CLK^			0.0	ns
51	$\overline{\text{E\_RDY}}$	hold wrt to CLK^			2.8	ns
52	E_AD[15:0]	ADDR setup to $\overline{\text{E\_AS}}$ ~~low~~ high			15.0	ns
53	E_AD[15:0]	ADDR hold wrt $\overline{\text{E\_AS}}$ high			0.0	ns
54	$\overline{\text{E\_HOLD}}$	setup to CLK^			0.0	ns
55	$\overline{\text{E\_HOLD}}$	hold wrt to CLK^			4.0	ns

Table 5.2 L64853 Switching Characteristics (Part 2 of 3)

No.	Signal	Description	Conditions	Min.	Max.	Units
56	Note 1	setup to CLK^		0.0		ns
57	Note 1	hold wrt to CLK^		3.0		ns
58	Note 2	hold wrt to CLK^		0.0		ns

Note 1: The values in Table 5.2 represent the timing characteristics of groups of signals. The timing diagrams show that one mnemonic value can represent many different signal paths.

Note 2: These timing parameters are true for both the signals $\overline{\text{D\_CS}}$ and $\overline{\text{ID\_CS}}$.

Note 3: The documented values represent the timing of an external device (in this case the ESP SCSI CHIP).

Note 4: The setup and hold times refer to the timing diagram on which they are shown, and in particular to the clock edges shown. Internal to the chip, the E_AD bus is not latched for at least 2 clock cycles to alleviate any potential timing problems; consequently, the 0 ns timing requirements shown are true only if the cycle-by-cycle handshaking specified by the LANCE is maintained.

Table 5.2 L64853 Switching Characteristics (Part 3 of 3)

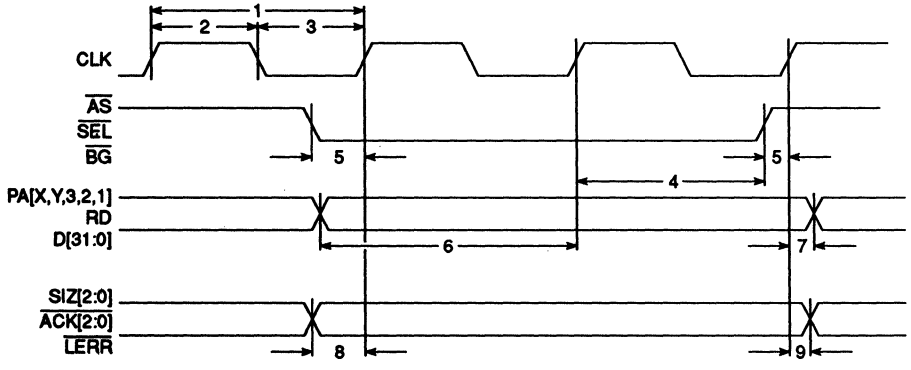


Figure 5.1 SBus Input Signals

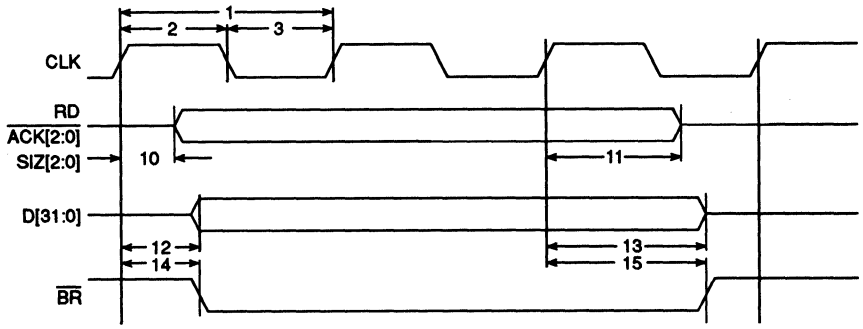


Figure 5.2 SBus Output Signals

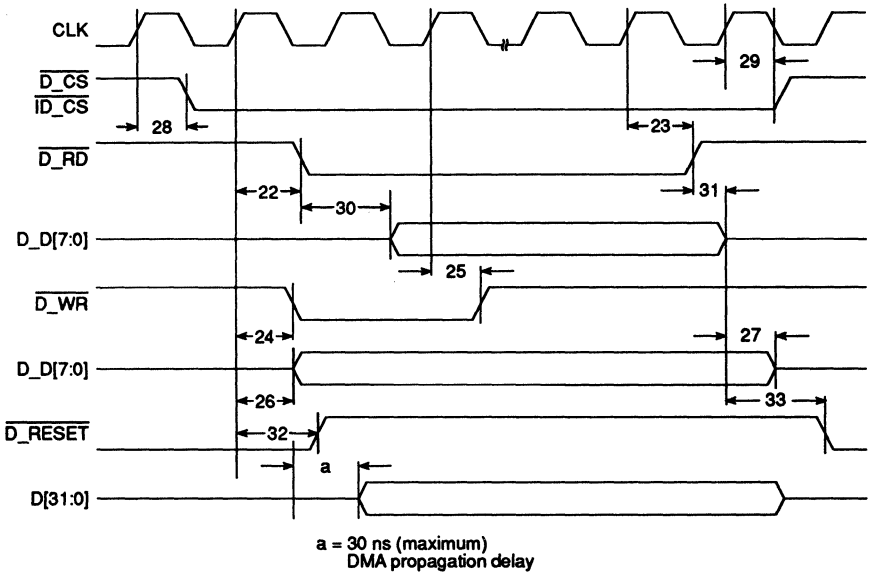


Figure 5.3 DMA Data Bus Read/Write Cycle Timing (Fast Cycle)

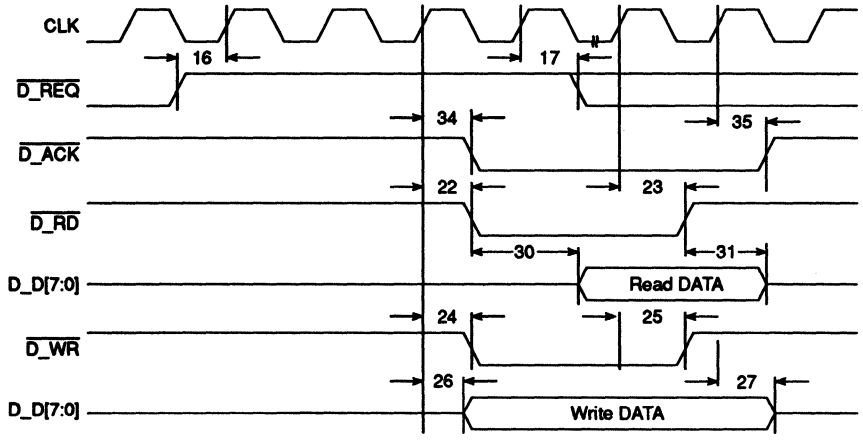
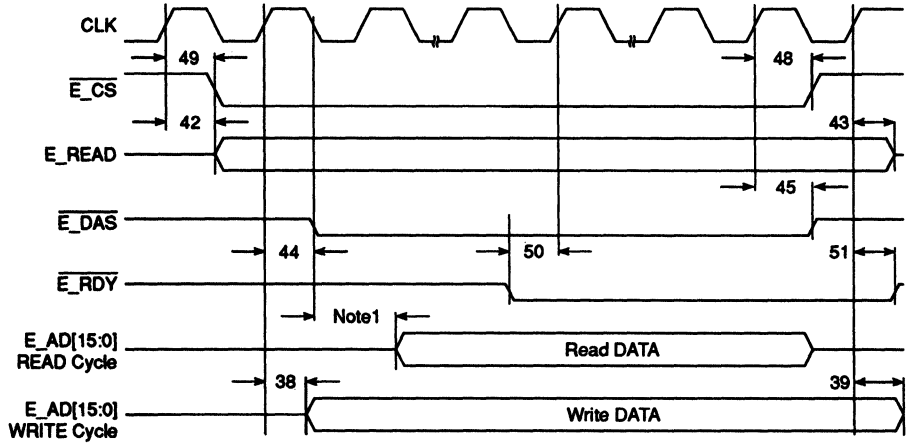
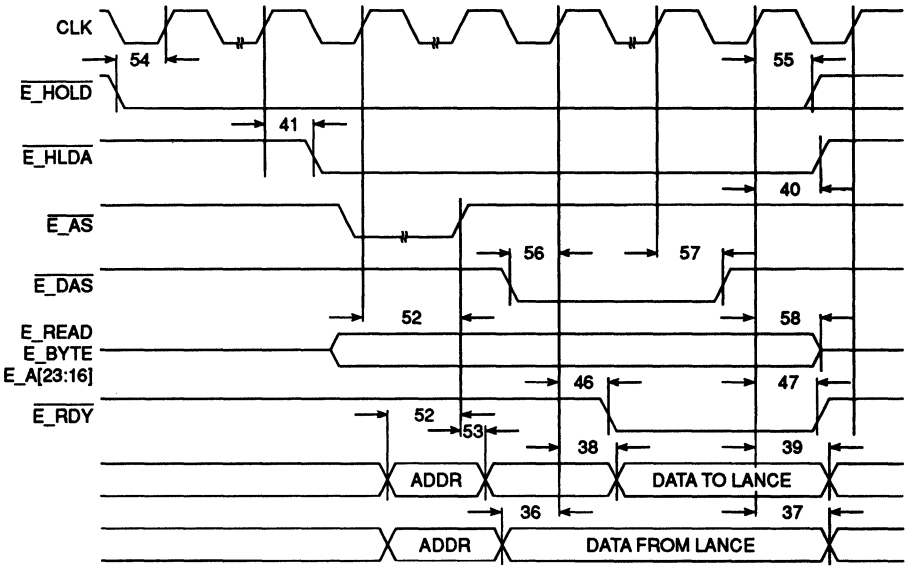


Figure 5.4 DMA Data Bus DMA Cycle Timing (Fast Cycle)



1. Refer to the AMD Am7990 LANCE Ethernet Controller manual or product specification for timing specifications.

Figure 5.5 LANCE Read/Write Cycle Timing



1. Refer to the AMD Am7990 LANCE Ethernet Controller manual or product specification for timing specifications.

Figure 5.6 LANCE DMA Read/Write Cycle Timing

5.2 DC Characteristics

This section specifies the DC electrical requirements for the L64853 DMA Controller. These requirements are listed in the following tables:

- Absolute Maximum Ratings
- Recommended Operating Conditions
- DC Characteristics

Parameter	Symbol	Limits	Unit
DC Supply Voltage	V_{DD}	-0.3 to +7	V
Input Voltage	V_{IN}	-0.3 to VDD +0.3	V
DC Input Current	I_{IN}	±10	µA
Storage Temperature Range	T_{STG}	-40 to +125	°C
References to V_{SS}			

Table 5.3 Absolute Maximum Ratings

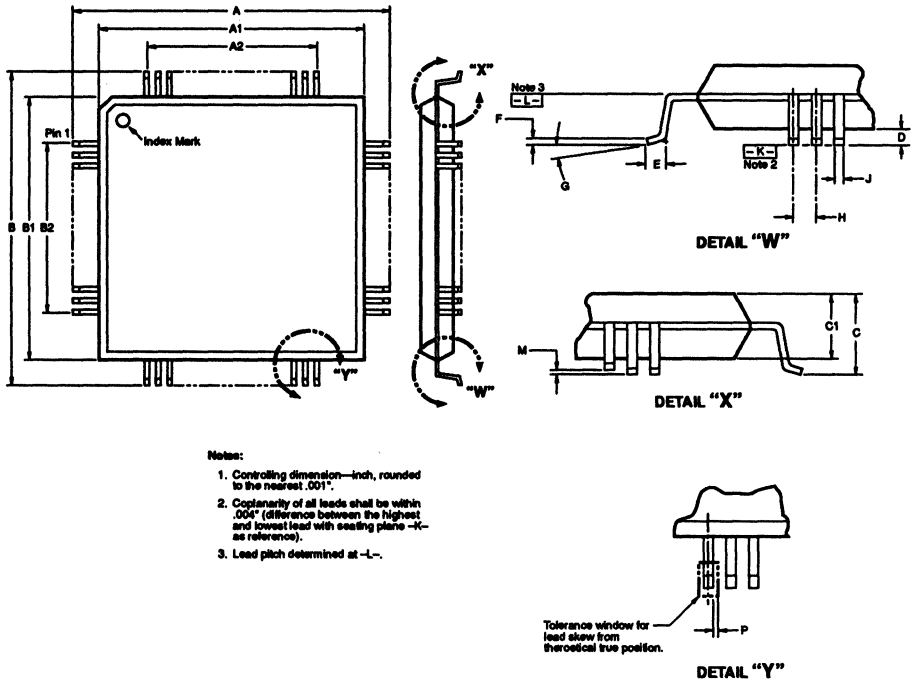
Parameter	Symbol	Limits	Unit
DC Supply Voltage	V_{DD}	0 to +5.25	V
Commercial (operating ambient temperature)	T_{STG}	0 to +70	°C

Table 5.4 Recommended Operating Conditions

Symbol	Parameter	Condition	Min	Typ	Max	Units
V_{IL}	Voltage Input LOW TTL inputs				0.8	V
V_{IH}	Voltage Input HIGH TTL Inputs				2.0	V
V_T+	Schmitt Trigger, Positive-Going Threshold			3.0	4.0	V
V_T-	Schmitt Trigger, Negative-Going Threshold		1.0	1.5		V
⊓	Hysteresis, Schmitt Trigger	V_{IL} to V_{IH} V_{IH} to V_{IL}	1.0	1.5		V
I_{IN}	Input Current, CMOS, TTL Inputs Inputs with Pulldown Resistors Inputs with Pullup Resistors	$V_{IN} = V_{DD}$ or V_{SS} $V_{IN} = V_{DD}$ $V_{IN} = V_{SS}$	-10 10 -100	±1 35 -30	10 120 -8	µA µA µA
V_{OH} (TTL)	Voltage Output HIGH	I_{OH}= -4.0 mA	2.4	4.5		V
V_{OL} (TTL)	Voltage Output LOW	I_{OL}= -4.0 mA		0.2	0.4	V
I_{OZ}	3-State Output Leakage Current	$V_{OH} = V_{DD}$ or V_{SS}	-10	±1	10	µA
I_{OS}	Current Open Short Circuit Current[1]	V_{DD} = Max, $V_O = V_{DD}$ V_{DD} = Max, $V_O = O_V$	15 -5	50 -25	130 -100	mA mA
C_{IN}	Input Capacitance	Any input[2]		2		pf
C_{OUT}	Output Capacitance	Any Output[3]		4		pf

[1]Not more than one output may be shorted at a time for a maximum duration of one second.
[2]Not applicable to assigned bidirectional buffer.
[3]Output using single buffer structure.

Table 5.5 DC Characteristics



Notes:

1. Controlling dimension—inch, rounded to the nearest .001".
2. Coplanarity of all leads shall be within .004" (difference between the highest and lowest lead with seating plane -K- as reference).
3. Lead pitch determined at -L-.

Dimensions		Inches (mm)
A	Min	1.244 (31.60)
	Max	1.276 (32.40)
A1	Min	1.098 (27.90)
	Max	1.106 (28.10)
A2	Ref	0.913 (23.20)
B	Min	1.244 (31.50)
	Max	1.276 (32.40)
B1	Min	1.098 (27.90)
	Max	1.106 (28.10)
B2	Ref	0.913 (23.20)
C	Max	0.152 (3.85)
C1	Max	0.138 (3.50)
D	Max	0.012 (0.30)
E	Min	0.024 (0.060)
	Max	0.039 (1.00)
F	Min	0.004 (0.10)
	Max	0.010 (0.25)
G	Min	0°
	Max	10°
H	Nom	0.31 (0.80)
J	Min	0.10 (0.25)
	Max	0.018 (0.45)
M	Max	0.004 (0.10)
P	Max	0.002 (0.05)

Figure 5.7 120-Pin PQFP Dimension Drawing

Chapter 6: Applications

This chapter provides information on using the L64853 DMA Controller in two types of SPARC Workstation environments and also describes how the controller can be configured with two peripheral chips: the AMD Am7990 LANCE Ethernet Controller and the Emulex SCSI Processor ESP-100. Both peripheral chips can be configured with the L64853 DMA Controller without additional circuitry to offer a low chip-count solution for networking and hard disk requirements.

For more information on these chips, refer to the appropriate documentation from Advanced Micro Devices, Inc. and Emulex Corporation, respectively.

6.1 The SBus and the SPARC Workstation

The L64853 has been developed for use with the SPARC family of chips. As such, it fits into one of two distinct kinds of SPARC Workstation environments: a host-based system and a symmetric system.

In a host-based system the CPU has a private address translation facility (in SBus terminology, this kind of system has a "CPU Master"). Host-based architectures are applicable in very high performance systems, such as the Sun SPARCstation 1, in which the SBus should be used only as a high-performance I/O interface and not as the CPU's channel to main memory. Figure 6.1 shows a typical host-based configuration.

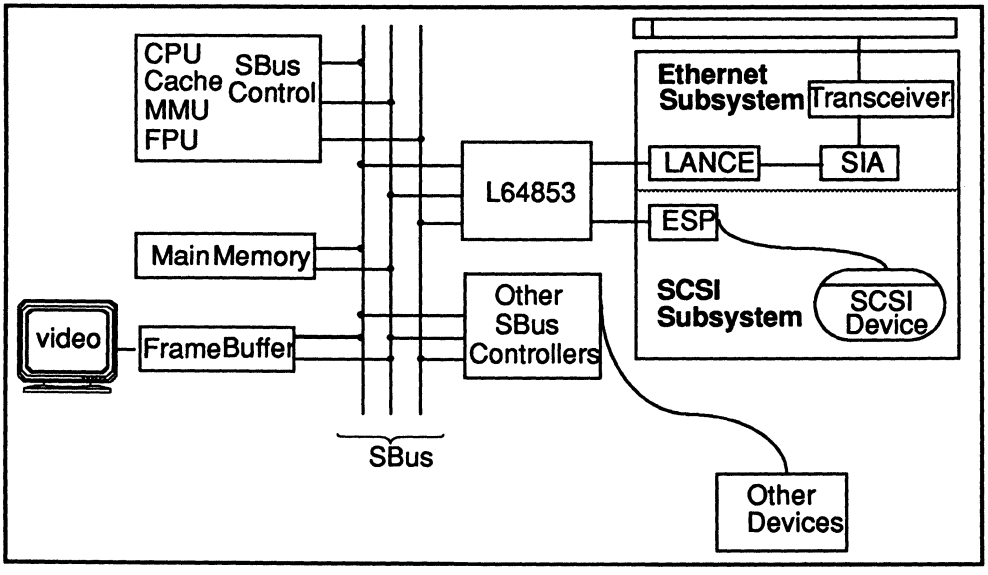


Figure 6.1 Host-Based System: SPARCstation 1

In a symmetric system, the CPU uses the same MMU as all the other SBus Masters (in SBus terminology, all SBus Masters in this kind of system are "DVMA Masters"). This type of architecture can be useful in low cost systems that don't need the extra circuitry that supplies the CPU with special address translation facilities. Figure 6.2 shows a symmetric system.

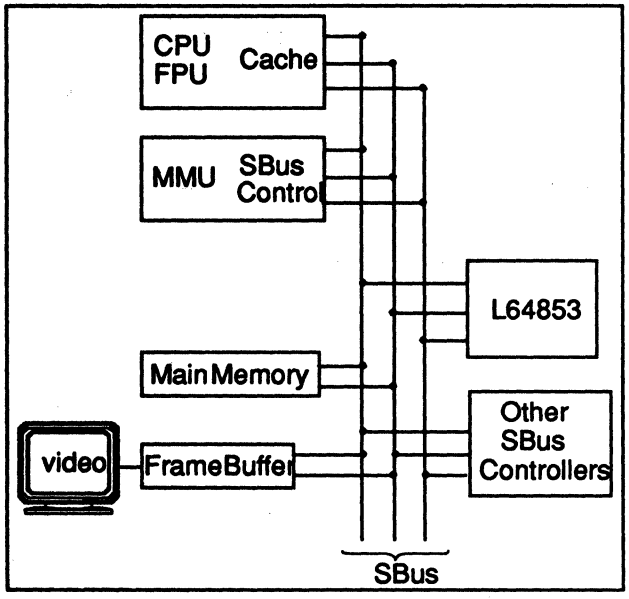


Figure 6.2 Symmetric Configuration

6.2 Using the L64853 With the EMULEX ESP100 SCSI Processor

This section illustrates the electrical interface for using the L64853 with the Emulex ESP100; it also shows how to access the ESP's internal registers.

Electrical Interface

The interface between the L64853 and the Emulex ESP100 is straightforward, and requires no additional logic. Figure 6.3 illustrates the interface.

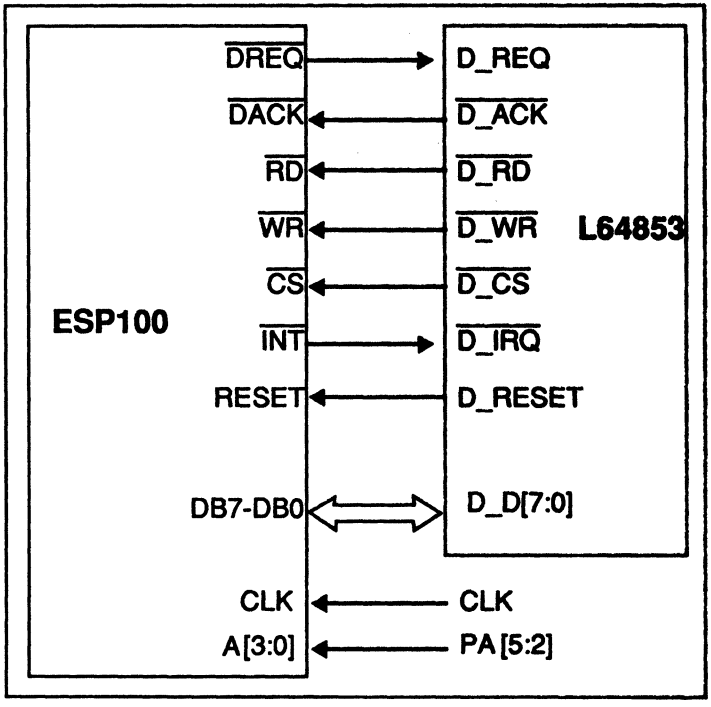


Figure 6.3 L64853 to ESP100 Interface

Figure 6.4 shows an ESP interface example using an external boot PROM.

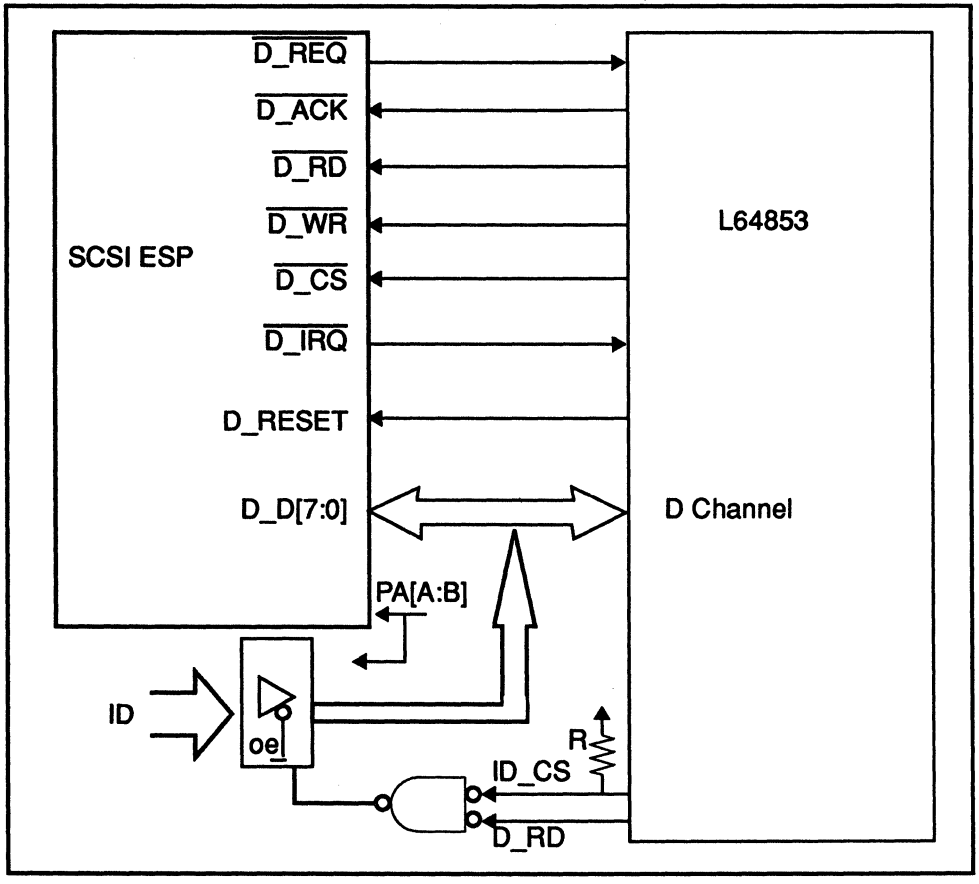


Figure 6.4 ESP Interface Example Using External Boot PROM

Accessing ESP Internal Registers

Table 6.1 shows how the physical address lines of the SBus on the SPARCstation 1 are wired to select the external registers of the ESP SCSI chip (note that "x" means either 1 or 0).

PA[X:Y] = PA[23:22]	PA[5:2]	PA1
10	0000	x Transfer Count Low
	0001	x Transfer Count High
	0010	x FIFO Data
	0011	x Command
	0100	x Status/Bus ID
	0101	x Interrupt/Status Timeout
	0110	x Seq.step/Synch transfer period
	0111	x FIFO flags/Synch offset
	1000	x Configuration
	1001	x Clock Conversion Factor
	1010	x ESP TEST
	1011	x Configuration
	11xx	x Reserved

Table 6.1 Physical Address Line Wiring on SBus for Selecting External Registers on ESP SCSI Chip

Notes:

1. The Enable Counter (bit 13) of the DMA Control/Status Register is not used with the ESP SCSI Controller.
2. The DMA Byte Counter Register is not used with the ESP SCSI Controller, which has its own Byte Counter external register.

6.3 Using the L64853 (With the AMD LANCE Ethernet Controller)

This section illustrates the electrical interface of the L64853 with the AMD Am7990 and shows how to access the LANCE's internal registers.

Electrical Interface

The interface between the L64853 and the LANCE is straightforward and requires no additional logic. Figure 6.5 illustrates the interface.

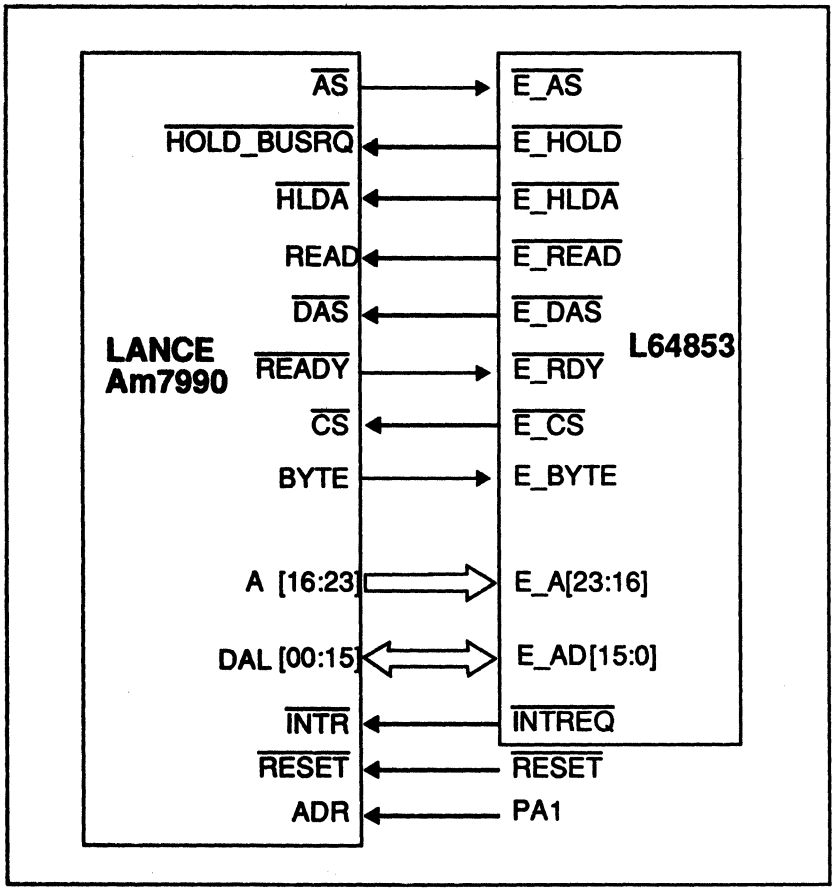


Figure 6.5 L64853 Am7990 LANCE Interface

Accessing LANCE Internal Registers

Table 6.2 shows how the physical address lines of the SBus on the SPARCstation 1 are wired up to select the external registers of the LANCE Ethernet chip (note that "x" means either 1 or 0).

PA[X:Y] = PA[23:22]	PA1
11	0 Register Data Port (RDP)

Table 6.2 Wiring of SBus Physical Address Lines for Selecting External Registers on LANCE ETHERNET Chip

Sales Offices and Design Resource Centers

LSI Logic Corporation
Headquarters
■ **1551 McCarthy Blvd**
Milpitas CA 95035
Tel: 408.433.8000
Telex: 171092
FAX: 408.434.8457

Alabama
■ 600 Boulevard South
Suite 104P
Huntsville, AL 35802
Tel: 205.883.3527
FAX: 205.883.3513

Arizona
8283 N Hayden Rd
Suite 270
Scottsdale AZ 85258
Tel: 602.951.4560
FAX: 602.951.4580

California
■ 2540 N. 1st Street
Suite 201
San Jose CA 95131
Tel: 408.954.1561
FAX: 408.954.1565

■ Two Park Plaza
Suite 1000
Irvine CA 92714
Tel: 714.553.5600
FAX: 714.474.8101

5625 Ruffin Rd
Suite 200
San Diego CA 92123
Tel: 619.541.7092
FAX: 619.541.2758

■ 15281 Ventura Blvd
Suite 275
Encino CA 91436
Tel: 818.379.2400
FAX: 818.783.5548

Colorado
3801 E Florida Ave
Suite 400
Denver CO 80210
Tel: 303.756.8800
FAX: 303.759.3486

Florida
201 Park Place
Suite 300
Altamonte Springs FL 32701
Tel: 407.339.2242
FAX: 407.831.3919

■ 1900 Glades Rd
Suite 201
Boca Raton FL 33431
Tel: 407.395.6200
FAX: 407.394.2865

Georgia
6525 The Corners Parkway
Suite 400
Norcross GA 30092
Tel: 404.448.4898
FAX: 404.449.9236

Illinois
■ One Pierce Place
Suite 400 East
Itasca IL 60143
Tel: 708.773.0111
FAX: 708.773.4631

Maryland
■ 6903 Rockledge Dr
Suite 230
Bethesda MD 20817
Tel: 301.897.5800
FAX: 301.897.8389

10480 Little Patuxent Pkwy
Suite 500
Columbia MD 21044
Tel: 301.740.5664
FAX: 301.740.2048

Massachusetts
■ 1601 Trapelo Rd
Waltham MA 02154
Tel: 617.890.0180 (Design Ctr)
Tel: 617.890.0161 (Sales Ofc)
FAX: 617.890.6158

Michigan
455 E Eisenhower Parkway
Suite 108
Ann Arbor MI 48108
Tel: 313.930.6975
Telex: 706456
FAX: 313.930.6978

Minnesota
■ 8300 Norman Center Drive
Suite 730
Minneapolis MN 55437
Tel: 612.921.8300
FAX: 612.921.8399

New Jersey
■ 379 Thornall St
Edison NJ 08837
Tel: 201.549.4500
FAX: 201.549.4802

New York
1065 Route 82
Hopewell Junction
New York NY 12533
Tel: 914.226.1620
FAX: 914.226.1315

North Carolina
■ 4601 Six Forks Rd
Suite 528, Phase 2
Raleigh NC 27609
Tel: 919.872.8400
FAX: 919.783.8909

Oregon
15455 NW Greenbrier Pkwy
Suite 210A
Beaverton OR 97006
Tel: 503.645.9882
FAX: 503.645.6612

Pennsylvania
Three Neshaminy Interplex
Suite 301
Trevose PA 19047
Tel: 215.638.3010
FAX: 215.245.4705

Texas
6034 W Courtyard
Suite 305
Austin TX 78730
Tel: 512.338.2140
FAX: 512.343.2612

■ 5080 Spectrum Drive
Suite 1010 West
Dallas TX 75248
Tel: 214.788.2966
FAX: 214.233.9234

Washington
■ 3015 112th Avenue NE
Suite 205
Bellevue WA 98004
Tel: 206.822.4384
FAX: 206.827.2884

LSI Logic Corporation
of Canada, Inc.
Headquarters
■ Petro-Canada Centre
#3410 150-6th Avenue SW
Calgary AB T2P 3Y7
Tel: 403.262.9292
FAX: 403.262.9494

■ 150 Karl Clark Road
Edmonton AB T6N 1E2
Tel: 403.450.4400
FAX: 403.450.4411

■ #306 3600 Gilmore Way
Burnaby BC V5G 4R8
Tel: 604.433.5705
FAX: 604.433.8443

■ #400 260 Hearst Way
Kanata ON K2L 3H1
Tel: 613.592.1263
Telex: 053.3849
FAX: 613.592.3253

■ #1110 401 The West Mall
Etobicoke ON M9C5J5
Tel: 416.620.7400
FAX: 416.620.5005

■ #600 755 St Jean Boulevard
Pointe Claire PQ H9S 5M9
Tel: 514.694.2417
FAX: 514.694.2699

France
LSI Logic S.A.
■ Tour Chenonceaux
204 Rond-point du Pont de Sevres
92516 Boulogne Billancourt
Tel: 33.1.46212525
Telex: 631475
FAX: 33.1.46203138

Israel
LSI Logic Limited
■ 40 Sokolov St
Ramat Hasharon 47235
Tel: 972.3.5403741/4
Telex: 371662
FAX: 972.3.5403747

Italy
LSI Logic SPA
■ Centro Direzionale Colleoni
Palazzo Orione, Ing. 1
20041 Agrate Brianza (MI)
Tel: 39.39.6056881
FAX: 39.39.653564

Japan
LSI Logic K. K.
■ Kokusai-Shin Akasaka
West Wing 13th Floor
6-1-20 Akasaka, Minato-Ku
Tokyo 108
Tel: 81.3.589.2711
FAX: 81.3.589.2740

■ 2-10-1 Kashuga
Tsukuba-Shi, Ibaraki 305
Tel: 81.298.52.8371
FAX: 81.298.52.8376

■ Twin 21 MID Tower 31st Floor
2-1-61 Shiromi, Higashi-Ku
Osaka 540
Tel: 81.6.947.5281
FAX: 81.6.947.5287

Tama-Plaza Daisen Bldg. 4th Floor
19-2, Utsukushigaoka 2 Chome
Mudor-Ku, Yokohama
Kanagawa-prf. 227
Tel: 81.45.902.4111
FAX: 81.45.902.4533

LSI Logic Corporation
of Korea Limited
■ 7th Floor 1304-3
Namseoul Building
Seocho-Dong
Seocho-Ku, Seoul
Tel: 82.2.561.2921
FAX: 82.2.554.9327

Netherlands
LSI Logic/Arcobel B.V.
■ Griekenweg 25
Postbox 344
NL-5340 AH Oss
Tel: 31.4120.30335
TWX: 37489
FAX: 31.4120.30635

Scotland
LSI Logic Limited
■ Lomond House
Beveridge Square
Livingstone EH54 6QS
Tel: 44.506.416767
FAX: 44.506.414836

Sweden
LSI Logic Export AB
TorShamnsgatan 39
S-16440 Kista
Tel: 46.8.703.4680
FAX: 46.8.7506647

Switzerland
LSI Logic/Sulzer Gmbh
■ Mittelstr. 24
CH-2580 Nidau/Biel
Tel: 41.32.515441
FAX: 41.32.516507

Taiwan
LSI Logic Corporation
■ 678 Tun Hua S. Rd
3rd Floor-2
Taipei, Taiwan R.O.C.
Tel: 02.755.3433
FAX: 02.755.5176

United Kingdom
LSI Logic Limited
■ Grenville Place
The Ring
Bracknell
Berkshire RG121BP
Tel: 44.344.426544
Telex: 848679
FAX: 44.344.481.039

West Germany
LSI Logic GmbH
Headquarters
■ Arabella Strasse 33
8000 Munich 81
Tel: 49.89.926903.0
FAX: 49.89.917096

■ Niederkasseler Lohweg 8
4000 Dusseldorf 11
Tel: 49.211.5961066
TWX: 8587248
FAX: 49.211.592130

■ Buechsenstrasse 15
7000 Stuttgart 1
Tel: 49.711.2262151
TWX: 723813
FAX: 49.711.2261124

AE Advanced Electronics
■ Stefan-George-Ring 19
8000 Munich 81
Tel: 49.89/93009855
FAX: 49.89/93009866

AE Advanced Electronics
■ Theaterstr. 14
3000 Hannover 1
Tel: 49.511/3681756
FAX: 49.511/3681759

■ Sales Offices with Design Resource Centers

Printed in USA
030.7111.6537.005.5K.LSI.D

LSI LOGIC

LSI Logic
Corporation
1551 McCarthy Blvd
Milpitas CA 95035
408.433.8000

Order No. L64853T Rev A
Part No. MD70-000109-99 A