PMAZ-AA TURBOchannel SCSI Module Functional Specification

Revision 1.2

Workstation Systems Engineering
Digital Equipment Corporation
100 Hamilton Avenue
Palo Alto, CA 94301

August 27, 1990

Blank Page

Table of Contents

Revision History

Date	Version	Content/Changes
04 Oct 89	1.0	Initial release
01 Nov 89	1.1	Cleaned-up version with Artemis drawings

Blank Page

1. PMAZ-AA SCSI Module

This is the functional specification for the PMAZ-AA SCSI Module. The SCSI module consists of the following subsystems.

- NCR's 53C94 Advanced SCSI Controller
- 128 KByte of SRAM buffer
- 32 KByte diagnostic ROM
- DMA address register

The SCSI module is a single-width slave only option card for TURBOchannel. The interface is based upon the NCR 53C94 Advanced SCSI Controller and 32K-by-32-bit (128 KBytes) SCSI buffer. The 53C94 can perform up to 5 MegaBytes/second asynchronous or synchronous data transfers on SCSI bus. The 53C94 has a command set that allows it to perform common SCSI sequences at hardware speed without host intervention. It has a on-chip FIFO, a 16-bit DMA interface and a 8-bit microprocessor interface which is configured to be shared with the lower half of DMA bus. This mode of operation is called the Single bus mode. The 53C94 performs DMA data transfers to and from the SCSI buffer 16 bits at a time. The buffer is time-multiplexed between the 53C94 and the processor. The processor fills the buffer before giving a go-ahead to the 53C94. The 53C94 performs DMA transfers to the SCSI buffer; it has no access to system memory.

The 53C94 also has on-chip 48 mA drivers for single-ended transmission. The maximum cable length supported is up to six meters. The SCSI cable is always terminated on the SCSI Option Module, which supplies termination power (through a current limit device) to the remote-end terminator. Up to seven SCSI peripherals may be connected to the cable.

2. Address Map

The DS5000/200 allocates the top 32 MBytes of physical address space to the I/O subsystems. The 32 MBytes I/O region is further divided into 8 regions of 4 MBytes each, for each one of the eight subsystems that may exist on the IO-bus. Refer to the *KN02 Functional Spec* for further details on the bus. Table 2-1 lists these regions.

Table 2-1: KN02 I/O Address Space

Address Range	Size (MBytes)	Sub-Systems
0x1FC00000..0x1FFFFFFF	4.0	System support
0x1F800000..0x1FBFFFFF	4.0	System module embedded I/O slot 6
0x1F400000..0x1F7FFFFF	4.0	System module embedded I/O slot 5
0x1F000000..0x1F3FFFFF	4.0	Reserved
0x1EC00000..0x1EFFFFFF	4.0	Reserved
0x1E800000..0x1EBFFFFF	4.0	Option Connector 2
0x1E400000..0x1E7FFFFF	4.0	Option Connector 1
0x1E000000..0x1E3FFFFF	4.0	Option Connector 0

The SCSI interface may reside in any of the three option connector slots. The 4 MByte address space on the SCSI module is further divided as shown in the Table 2-2

Table 2-2: SCSI Module Address Map

Address Range	Subsystem
0xbb+00000..0xbb+3FFFF	53C94 Registers
0xbb+40000..0xbb+7FFFF	DMA Address Register
0xbb+80000..0xbb+BFFFF	SCSI SRAM Buffer
0xbb+C0000..0xbb+FFFFF	SCSI Diagnostic ROM

The SCSI subsystem does not occupy the entire 4 MByte region. References to portions of SCSI subsystem address space not explicitly defined in the following sections should not be issued. "bb" is the base address of the I/O slot in which the SCSI interface resides.

The R3000 CPU is configured for little-endian byte order. All address space descriptions in this document are correspondingly little-endian.

3. Interrupts

Table 3-1 lists the I/O device connections to the R3000 interrupt inputs. The state of the interrupt signals is continually reflected in the R3000 CAUSE register at the bit position shown in the table. Note that a given interrupt signal only generates an R3000 exception if it is enabled in the STATUS register interrupt mask field, and interrupts are enabled by the STATUS<0> register bit. Note also that the interrupt signals are visible in the CAUSE register regardless of the state of the STATUS interrupt mask. That is, the operating system interrupt dispatcher must explicitly check that a given interrupt level, which is asserted in the CAUSE register, is enabled before activating that interrupt level's handler.

Table 3-1: I/O Interrupt Levels

Level	CAUSE/STATUS	Source
5	15	FPU
4	14	Reserved
3	13	Memory Error
2	12	Reserved
1	11	Real Time Clock
0	10	I/O Slots

Due to the increased number of I/O devices supported in TURBOchannel systems, all I/O interrupts are merged into a single R3000 interrupt. However, the individual I/O device interrupts are visible through the system module interrupt status register. The bits in the register are as shown in the Table 3-2

Table 3-2: Local Interrupt Status Register

Bit	I/O Device
7	Serial Lines
6	System module embedded I/O slot 6
5	System module embedded I/O slot 5
4	Reserved
3	Reserved
2	Option Connector 2
1	Option Connector 1
0	Option Connector 0

The 53C94 will interrupt the CPU through the Local interrupt at level 0, which will be visible on the R3000 CAUSE<10> register bit. The host should then read the Local Interrupt Status register to determine which I/O device is interrupting.

4. Power Estimates

Table 4-1 lists the power consumption estimates for the SCSI module.

Table 4-1: Power Estimates

Supply(V)	Typical Current (A)	Max Current (A)
+5	1.25	1.9

5. External Interface

The SCSI module connects to the SCSI bus via a DB50 receptical type connector. Table 5-1 lists the pinout of the connector.

Table 5-1: SCSI connector pinout

Pin	Signal	Pin	Signal	Pin	Signal	Pin	Signal
1	gnd	14	gnd	26	~sd0	39	gnd
2	gnd	15	gnd	27	~sd1	40	gnd
3	gnd	16	gnd	28	~sd2	41	~atn
4	gnd	17	gnd	29	~sd3	42	gnd
5	gnd	18	gnd	30	~sd4	43	~bsy
6	gnd	19	gnd	31	~sd5	44	~ack
7	gnd	20	gnd	32	~sd6	45	~rst
8	gnd	21	gnd	33	~sd7	46	~msg
9	gnd	22	gnd	34	~sdp	47	~sel
10	gnd	23	gnd	35	gnd	48	~cd
11	gnd	24	gnd	36	gnd	49	~req
12	gnd	25	gnd	37	gnd	50	~io
13	nc			38	trmPwr		

6. Functional Description

6.1. 53C94

The 53C94 supports half-word reads and writes in DMA mode and byte reads and writes in the programmed I/O mode. The 53C94 registers are word-aligned. Reads of the 54C94 nominally stall the CPU for 10 cycles. Writes to the 53C94 nominally complete at a rate of 6 cycles per byte. In DMA mode the 53C94 can perform reads and writes of the SCSI buffer at a rate of 6 cycles per half-word, a peak transfer rate of 8.33 MBytes/sec.

Table 6-1 lists the 53C94 register addresses. Refer to *53C94 Functional Description* for further details about the chip.

Table 6-1: 53C94 Register Addresses

Address	Name	RD/WR	Register
0xbb+00000	ASC_TCLSB	R/W	Transfer Counter LSB
0xbb+00004	ASC_TCMSB	R/W	Transfer Counter MSB
0xbb+00008	ASC_FIFO	R/W	FIFO
0xbb+0000C	ASC_CMD	R/W	Command
0xbb+00010	ASC_STAT	R	Status
0xbb+00010	ASC_DBID	W	Destination Bus ID
0xbb+00014	ASC_INTR	R	Interrupt
0xbb+00014	ASC_SRTO	W	Select/Re-select Timeout
0xbb+00018	ASC_SS	R	Sequence Step
0xbb+00018	ASC_SP	W	Synchronous Period
0xbb+0001C	ASC_FFSS	R	FIFO Flags/Sequence Step
0xbb+0001C	ASC_SO	W	Synchronous Offset
0xbb+00020	ASC_CNF1	R/W	Configuration 1
0xbb+00024	ASC_RVSD	R	Reserved
0xbb+00024	ASC_CCF	W	Clock Conversion Factor
0xbb+00028	ASC_RVSD	R	Reserved
0xbb+00028	ASC_TM	W	Test Mode
0xbb+0002C	ASC_CNF2	R/W	Configuration 2
0xbb+00030	ASC_CNF3	R/W	Configuration 3

6.2. SCSI Buffer

The SCSI buffer is a 32K-by-32-bit SRAM buffer and supports byte, half-word, tribyte, and word reads and writes from the processor side. The ~bm[3..0] field in the I/O bus is latched with the address and only the appropriate bytes are written into the SCSI buffer. Upon reads the whole word is supplied, and the ~bm field is ignored, with the R3000 performing the appropriate byte masking and alignment. From the 53C94 side the SCSI buffer supports half-word reads and writes in the DMA mode.

Reads of the buffer nominally stall the CPU for 8 cycles; a peak read bandwidth of 12.5 MBytes/second. Writes to the buffer nominally complete at a rate of 5 cycles per word; a peak write bandwidth of 20 MBytes/second. Buffer access during 53C94 activity may increase the access latency by up to 6 additional cycles.

6.3. DMA Address Register

The DMA Address Register is located at 0xbb+40000. This register is a write only register. It is written by the CPU with the start DMA address and the DMA direction for the 53C94 to use. The register should be written before giving a DMA transfer command to the 53C94. The address loaded into the register should be half-word-aligned, byte address; bit <0> is ignored. Table 6-2 shows the bit assignments of this register.

Table 6-2: DMA Address Register

Bits	Function
31	DMA Write
30..17	Not Used
16..0	Start DMA Address

Bit <31> asserted (one) means a DMA write to the 53C94. DMA write means that the external DMA controller will read the data out of the RAM buffer and write into the 53C94. Bit <31> deasserted (zero) means a DMA read from the 53C94. DMA read means that the external DMA controller will read the data out of the 53C94 and write into the RAM buffer. A write to this register will nominally finish in 7 cycles and a read from this register will not return ready, thus causing a timeout on the bus.

6.4. SCSI Diagnostic ROM

The SCSI Diagnostic ROM is located at 0xbb+C0000. The ROM is 32K by 8-bits and is word-aligned. The data is presented on the least significant byte of the data bus. A write to this address space will not return ready, thus causing a timeout on the bus. A read of the ROM will nominally stall the CPU for 12 cycles.

7. 53C94 FUNCTIONAL DESCRIPTION

The ASC has a command set that allows it to perform common SCSI sequences at hardware speed without host intervention. Its on-chip FIFO may be accessed simultaneously by the SCSI bus and either the microprocessor or the host DMA controller. All Command, Data, Status and Message bytes pass through the FIFO on their way to or from the SCSI bus. Most ASC commands have two versions: DMA and non-DMA. When DMA instructions are used, data will pass between memory and the SCSI bus with the FIFO acting as temporary storage when the DMA channel is temporarily shut down by a higher priority event, such as DRAM refresh.

The FIFO also helps speed execution during non-DMA transfers. For example, in Initiator role, the microprocessor will load the Command Descriptor Block and optionally, one or three Message bytes into the FIFO, issue one of several Selection commands and wait for an interrupt. The ASC will wait for Bus Free; Arbitrate for the bus--again and again until it acquires it; send the Message bytes, followed by the Command Descriptor Block; then generate an Interrupt. Meanwhile, a multi-tasking host may continue with other tasks.

In Target role, the microprocessor will Enable Selection, then wait for an interrupt. Eventually, an Initiator will select the ASC. It will then automatically step through the Selection and Command Phases before generating an interrupt. When the interrupt occurs, the entire Command Descriptor Block will be in the FIFO along with any Message bytes sent by the Initiator. Combination commands, such as these, are identified with the *sequence* suffix in the table of ASC commands.

After Selection Phase has been successfully completed, the ASC may transfer bytes in any of the SCSI Information Phases whether operating in Initiator or a Target role. The ASC supports Disconnect-ReSelect in both Initiator and Target roles--making high performance multi-threaded systems easy to implement.

The ASC may transfer Data Phase bytes across the bus synchronously, at speeds up to 5 MB/s, or asynchronously at speeds up to 6 MB/s. See the section *Data Transfer Rate*. The difference between the two is transparent to the user except that the Synchronous Offset and the Synchronous Transfer Period registers must be programmed prior to synchronous data transfer. The default, after hardware or software reset is asynchronous transmission.

Data Phase bytes will usually be transferred using DMA. The microprocessor will program an external DMA controller, program the ASC Transfer Count, issue an ASC data transfer command (there are several), then wait for an interrupt. The DMA controller and the ASC will transfer all the data without microprocessor intervention.

To end the SCSI transaction, the ASC Target will place a Status byte and a Message byte in the FIFO; then issue a single command (there are two to chose from) which will cause the ASC to first assert Status Phase, send the first byte, assert Message In Phase, send the second byte, disconnect from the SCSI bus (after the Initiator releases ACKnowledge) and interrupt the microprocessor.

The end of a SCSI transaction is similar for an ASC Initiator except that it receives two bytes into its FIFO. The Initiator prevents the Target from disconnecting by holding ACKnowledge asserted on the bus while the microprocessor examines the Status and Message bytes. If both bytes are good, the Message Accepted command is used to instruct the ASC to release ACK, which allows the Target to disconnect which causes the Initiator to interrupt its host and report the disconnect. If the Status and Message bytes are not good, the host should first issue the Set ATN command before issuing the Message Accepted command. This instructs the ASC to assert ATN (Attention) before releasing ACK, which should cause the Target to request Message Out Phase rather then disconnect.

7.1. Bus Initiated Sequences

- Selection
- ReSelection
- SCSI Bus Reset

Selection or ReSelection sequences occur in the disconnected state when the ASC is Selected or ReSelected by another Initiator or Target--if the Enable Selection or ReSelection Command had previously been received by the ASC.

In addition to responding to bus initiated events, the ASC may initiate a bus event by using one of several Selection or ReSelection Commands. If one of these commands starts executing, **it will clear Enable Selection/ReSelection** after Arbitration has been won. Normally the microprocessor will have 250 ms (ANSI recommended Selection Time-Out period) after the chip disconnects from the bus to re-enable bus initiated events. If the time-out is exceeded, an Initiator or Target which is attempting to connect to the ASC, may time-out and abort.

If, on the other hand, the bus initiated event occurs before the command starts executing, the FIFO will be cleared, the Command Register will be cleared, and any further writes by the microprocessor will be ignored until the Interrupt Register is read. Since a Selection or ReSelection command requires that something be placed in the FIFO, these bytes will be lost--as will any command written to the Command Register. The interrupt handler that services a Selection or ReSelection command will have to examine the bits in the Interrupt register to determine if the ASC selected another device, or if it was selected by another device. The former case will cause a Function Complete interrupt, the latter case will cause a Selection or ReSelection interrupt.

7.1.1. Bus Initiated Selection

When the ASC has been selected as a Target, the following data will be in its FIFO:

- Bus ID
- Identify Message
- Command Descriptor Block

The Bus ID will always be present and will always be one byte. It is an un-encoded version of the state of the bus during Selection Phase. Any SCSI data bits that were true during Selection Phase will be set. The Target ID (our ID) must always be set. In Arbitrating systems, the Initiator ID must also be set. The Initiator ID is optional in non-arbitrating systems.

The Identify Message will always be placed in the FIFO and will always be one byte in SCSI-1 systems but may be one or three bytes in SCSI-2 systems. If the ASC is selected with ATN false, it will store a null byte (00) in the FIFO behind the Bus ID, then begin requesting Command Phase bytes. A detected parity error will cause the ASC to interrupt and stop.

If the ASC is selected with ATN true, and the SCSI-2 bit is not set, it will request one Message byte, place it in the FIFO behind the Bus ID, then begin requesting Command Phase bytes unless the Message byte is not a valid Identify Message, or a parity error is detected--which will cause the ASC to interrupt and stop. The Sequence Step Register should then be examined.

If the ASC is selected with ATN true and the SCSI-2 bit is set, the ASC will examine both the Message byte and the ATN signal to determine how many bytes to request. If the first byte is a valid Identify Message and if ATN goes false after receiving the first byte, the ASC will only request one message byte. If the first byte is a valid Identify Message byte and ATN is still true, it will request two more Message bytes. The ASC will then begin requesting Command Phase bytes unless the first byte was not a valid Identify Message, or a parity error was detected, or ATN went false between the 2nd and 3rd bytes, or ATN remained true but the SCSI-2 bit was false; which will cause the ASC to interrupt and stop. The Sequence Step Register should then be examined.

The Command Descriptor Block (CDB) will always begin at the third or fifth byte in the FIFO, assuming Selection completed normally. The CDB may be 6, 10 or 12 bytes long. Thus, in SCSI-2, the entire FIFO may be filled if a tagged queue twelve byte command is used.

7.1.2. Bus Initiated ReSelection

The ASC will allow itself to be ReSelected as an Initiator by a Target if it has previously received the Enable Selection/ReSelection command. If the sequence completes normally, the following information will be in the FIFO:

- Bus ID
- Identify Message

The Bus ID is the same as the Selection case, described above. The Identify Message will always be present and always be one byte.

7.1.3. Bus Initiated Reset

A bus initiated reset will be recognized by the ASC at any time. The ASC will then disconnect from the bus and reset its internal sequencer. If the SCSI Reset Reporting/Interrupt Disable bit (Configuration -1 Register) is not set the ASC will generate a SCSI Reset Detected interrupt.

7.2. Parity Checking and Generation