STANDARD MICROSYSTEMS CORPORATION
35 Marcus Blvd., Hauppauge, N.Y. 11788
(516) 273-3100 • FAX: 516-231-6004

MSD95C01/02

T-52-33-63

Storage μ-Controller for Direct Access (Disk) or Serial Access (Tape) Devices — SμnDAe™

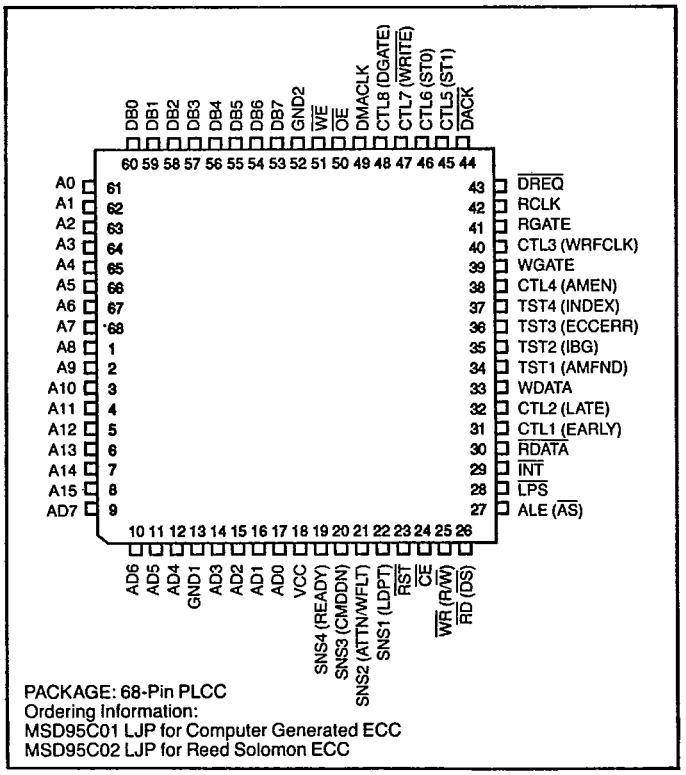
APPLICATIONS SUPPORTED

- ☐ Embedded SCSI Drives When Used With Companion MSD95C00 SCSI Controller
- ☐ IBM PS/2 ESDI Adapter When Used With Companion MSD95C10 Micro Channel ESDI Interface Chip
- ☐ IBM AT Adapter When Used With Companion MSD95C15 AT Interface Chip
- ☐ Compatible With ESDI and SMD Disk Interfaces
- ☐ Interfaces to 8051, Z8 and 80188 Microprocessors
- ☐ Supports 3-Sector Prefetch for Unix® Applications
- ☐ Supports QIC-24 Tape:
  - 5 Mb/sec GCR Data Transfers
  - Read-After-Write
- ☐ Controls Optical Disks and Floppy Disks
- ☐ Modular Design for Easy Adaptation to Special Purpose Applications

FEATURES

- ☐ 24 Mb/sec NRZ or 12 Mb/sec RLL, MFM, FM Disk Data Transfers
- ☐ Zero Latency Read Capability
- ☐ Choice of ECC: 32-bit IBM® Compatible ECC and one of the following:
  - –(MSD95C01): Computer Generated ECC Fully Programmable from One to 64 Bits in One Bit Increments.
  - or
  - –(MSD95C02): Extended Reed-Solomon ECC Able to Detect and Correct a 41 Bit Burst Error or Two Randomly Spaced 17 Bit Burst Errors.
- ☐ Register Programmable Data Format via On-Chip Writeable Microsequencer
- ☐ Supports Transparent "On-The-Fly" Error Correction

PIN CONFIGURATION



PACKAGE: 68-Pin PLCC
Ordering Information:
MSD95C01 LJP for Computer Generated ECC
MSD95C02 LJP for Reed Solomon ECC

- ☐ 3-Channel Internal Double-Speed 64K (Externally Expandable) Ring Buffer DMA Controller
- ☐ Cache Buffer Management Allows Disk Data Transfers Without Processor Intervention
- ☐ Low Power CMOS with Standby Mode

GENERAL DESCRIPTION

The MSD95C01/02 is a high speed micro-programmable data path controller. It incorporates a triple channel DMA CONTROLLER, a RAM based MICROSEQUENCER, a sophisticated ECC generator/checker circuit, an RLL2, 7/MFM/FM/GCR Encoder/Decoder and a Parallel/Serial shift register in one 68 pin plastic package.

The MSD95C01/02 is built from a set of high level, function specific SuperCells™ that can be connected together in such a way as to adapt themselves to a special purpose customer requirement. The MSD95C01/02 can be combined in a circuit with standard local processor and static RAM chips to build a very high performance multi-media controller incorporating SCSI, ESDI, SMD, ST-506, QIC 24, and FLOPPY disk interfaces. The RAM based MICROSEQUENCER permits the user to build a mass storage controller that conforms with any currently available data format.

The addition of an SMC MSD95C00 SCSI CONTROLLER will provide a tightly coupled 2-chip set for high speed, high performance SCSI and "Embedded SCSI" applications with minimum component count and synchronous SCSI speeds of up to 5 megabytes per second.

Combining the MSD95C01 with the SMC MSD95C10 MICRO CHANNEL INTERFACE DEVICE will result in a highly integrated ESDI disk interface for the IBM Micro Channel Bus. The chip set may be used for adapter boards as well as PS/2 motherboard applications.

The SMC MSD95C15 IBM AT INTERFACE DEVICE can be used with the MSD95C01 DISK CONTROLLER to build an IBM AT hard disk interface for MFM, RLL and ESDI drives.

TABLE OF CONTENTS

DESCRIPTION OF PIN FUNCTIONS

T-52-33-63

PIN NO.	NAME	SYMBOL	DESCRIPTION
RING BUFFER INTERFACE			
53-60	Data Bus	DB7-0	Input/Output. Bi-directional data bus to the external ring buffer RAM. This bus is automatically put into a high impedance state during valid External Device DMA cycles.
61-68, 1-8	Address Bus	A0-A15	Output. These signals are used to address the external ring buffer RAM.
50	Output Enable	$\overline{OE}$	Output. This active low signal strobes the ring buffer RAM's data bus output drivers.
51	Write Enable	$\overline{WE}$	Output. This active low signal strobes write data from the data bus into the ring buffer RAM.
49	DMA Clock	DMACLK	Input. This 20 MHz (maximum) signal is used by the MSD95C01/02 to generate DMA cycles.
43	DMA Request	$\overline{DREQ}$	Input. This signal is driven active low by an External Device to request a DMA cycle.
44	DMA Acknowledge	$\overline{DACK}$	Output. This signal is driven active low by the MSD95C01/02 in response to $\overline{DREQ}$. During data transfer from the ring buffer to the External Device, the rising edge of this signal is used to strobe data into the External Device. During data transfer from the external device to the ring buffer, a low will enable data transfer from the External Device.
45 46	Control 5 Control 6	CTL5 (ST1) CTL6 (ST0)	Output. Depending on the programming of the MODE 1 Register, these bits can either reflect the data written to bits 4 and 5 of the Local Processor Output Register or they can be outputs that indicate the type MSD95C01/02 DMA controller cycle in progress: ST1 ST0 CYCLE 0 0 EXTERNAL DEVICE 0 1 LOCAL PROCESSOR 1 0 MICROSEQUENCER (DISK) 1 1 RESERVED
47	Control 7	CTL7 ($\overline{WRITE}$)	Output. Depending on the programming of the MODE 1 Register, this bit can either reflect the data written to bit 6 of the Local Processor Output Register or this bit can be an output signal indicating to the ring buffer that a WRITE or READ cycle to or from the ring buffer is about to start. This output is low during a write cycle and high during a read cycle. CTL 7, 6, and 5, when programmed as $\overline{WRITE}$, ST0, ST1, can be used for interfacing to an external ECC chip.
48	Control 8	CTL8 (DGATE)	Output. Depending on the programming of the MODE 1 Register, this bit can reflect the data written to bit 7 of the Local Processor Output Register. It may also be used as an output of the MICROSEQUENCER to be used as a "data valid" signal to indicate when data is being transferred on the DB7-0 bus for interface with an external ECC chip.
DRIVE INTERFACE			
30	$\overline{Read\ Data}$	$\overline{RDATA}$	Input. This signal is the serial, active low data from the disk or tape drive. It may be encoded as RLL2, 7, MFM, FM, GCR, or NRZ as selected in MODE 2 Register bits 2-0.
41	Read Gate	RGATE	Output. This signal is typically used to enable the data separator to begin locking to data. Normally, it becomes active during the PLO sync field. This signal is controllable via microcode to allow specific read data search algorithms and conformance to unique drive formats.
42	Read Clock (Read/ Reference Clock)	RCLK	Input. This clock is used to frame the encoded $\overline{RDATA}$ bit stream from the drive. For NRZ input data, the Read/Reference Clock signal provides the timing necessary to synchronize the serial data transfer between the drive and the MSD95C01/02. RCLK is divided internal to the MSD95C01/02 and is used to run the MICROSEQUENCER. It is therefore necessary to provide a glitch-free clock into this input when the microprocessor is in operation.
33	Write Data	WDATA	Output. This signal is the serial NRZ, FM, MFM, GCR or RLL data being written to the drive.
39	Write Gate	WGATE	Output. This signal is controlled by the microsequencer and is active when the MSD95C01/02 is writing data to the drive.

DESCRIPTION OF PIN FUNCTIONS (CONTINUED)

T-52-33-63

PIN NO.	NAME	SYMBOL	DESCRIPTION
DRIVE INTERFACE (CONTINUED)			
31	Control 1	CTL1 (EARLY)	Output. Depending on the programming of the MODE 3 Register, this bit either reflects the data written to bit 0 of the Local Processor Output Register or indicates that the current disk Write Data bit should be externally precompensated early.
32	Control 2	CTL2 (LATE)	Output. Depending on the programming of the MODE 3 Register, this bit either reflects the data written to bit 1 of the Local Processor Output Register or indicates that the current disk Write Data bit should be externally precompensated late.
40	Control 3	CTL3 (WRFCLK)	Input/Output. Depending on the programming of the MODE 3 Register, this bit either reflects the data written to bit 2 of the Local Processor Output Register or acts as a tape write reference clock input. In tape applications that require read after write capability, this pin must be programmed as an input (WRFCLK).
38	Control 4	CTL4 (AMEN)	Output. Depending on the programming of the MODE 2 Register, this bit either reflects the data written to bit 3 of the Local Processor Output Register or acts as a MICROSEQUENCER output which may be used to write an Address Mark (WGATE active) or search for an address mark (WGATE, RGATE inactive) in ESDI drive applications.
19	Sense Input 4	SNS4 (READY)	Input. The MSD95C01/02 can be programmed to generate a high-to-low level change interrupt from this pin. This input may be used to sense READY status from the drive.
20	Sense Input 3	SNS3 (CMDDN)	Input. The MSD95C01/02 can be programmed to generate a high-to-low level change interrupt from this pin. This input may be used to indicate a Command Complete status when using ESDI drives.
21	Sense Input 2	SNS2 (ATTN/WFLT)	Input. The MSD95C01/02 can be programmed to generate a high-to-low level change interrupt from this pin. This input may be used to sense ATTENTION status from an ESDI drive or a WRITE FAULT status from an ST-506 drive.
22	Sense Input 1	SNS1 (LDPT)	Input. The MSD95C01/02 can be programmed to generate any level change interrupt from this pin. This input may be used to sense load point status for tape applications.
37	Test Input 4	TST4 (INDEX)	Input. This edge-triggered signal is used by the MICROSEQUENCER for conditional branching. Typically, the INDEX pulse from the drive is connected to this pin.
36	Test Input 3	TST3 (ECCERR)	Input. This level-triggered signal is used by the MICROSEQUENCER for conditional branching. When used with an external ECC chip, this input may be used to indicate an ECC error.
35	Test Input 2	TST2 (IBG)	Input. This level-triggered signal is used by the MICROSEQUENCER for conditional branching. An external signal indicating interblock gap for tape applications may be connected to this pin, but if it is not connected, then this pin must be grounded in tape applications.
34	Test Input 1	TST1 (AMFND)	Input. This edge-triggered signal is used by the MICROSEQUENCER for conditional branching. Typically, an ESDI address mark found signal is connected to this pin.
PROCESSOR INTERFACE			
9-12, 14-17	Address/ Data Bus	AD7-0	Input/Output. Multiplexed bi-directional address/data bus to local processor.
27	Address Latch Enable (Address Strobe).	ALE ($\overline{AS}$)	Input. This signal is active when an address is valid on the AD7-0 bus. The local processor reads the MSD95C01/02 RESET Register address either following a hard or prior to a soft reset to automatically configure the MSD95C01/02 to expect ALE or $\overline{AS}$ at this input.
29	$\overline{\text{Interrupt}}$	$\overline{INT}$	Output. This open collector signal is driven low when the MSD95C01/02 detects an enabled interrupt.
26	$\overline{\text{Read}}$ Strobe ($\overline{\text{Data}}$ Strobe)	$\overline{RD}$ ($\overline{DS}$)	Output. When the MSD95C01/02 is configured for ALE, this active low strobe is used to enable read data from the MSD95C01/02 onto the AD7-0 bus. When the MSD95C01/02 is configured for $\overline{AS}$, this active low signal is used to strobe data into or out of the MSD95C01/02.

DESCRIPTION OF PIN FUNCTIONS (CONTINUED)

T-52-33-63

PIN NO.	NAME	SYMBOL	DESCRIPTION
PROCESSOR INTERFACE (CONTINUED)			
25	Write Strobe (Read/Write)	WR (R/W)	Input. When the MSD95C01/02 is configured for ALE, this active low strobe is used to latch write data from the AD7-0 bus into the MSD95C01/02. When the MSD95C01/02 is configured for AS, the R/W input is used to qualify DS for a read or write cycle.
24	Chip Enable	CE	Input. This input signal is used to qualify the RD and WR strobes for all accesses on the AD7-0 bus. This signal must be valid throughout the memory cycle.
28	Low Power Standby	LPS	Input. A low level applied to this pin signifies that the system is requesting the low power standby mode. This pin should be tied to a logic "1" for normal operation.
23	Reset	RST	Input. A low level applied to this pin will cause the MSD95C01/02 to be reset to a known state.
18	Power Supply	VCC	+5V Power Supply pin.
13	Ground 1	GND1	Ground pin.
52	Ground 2	GND2	Ground pin.

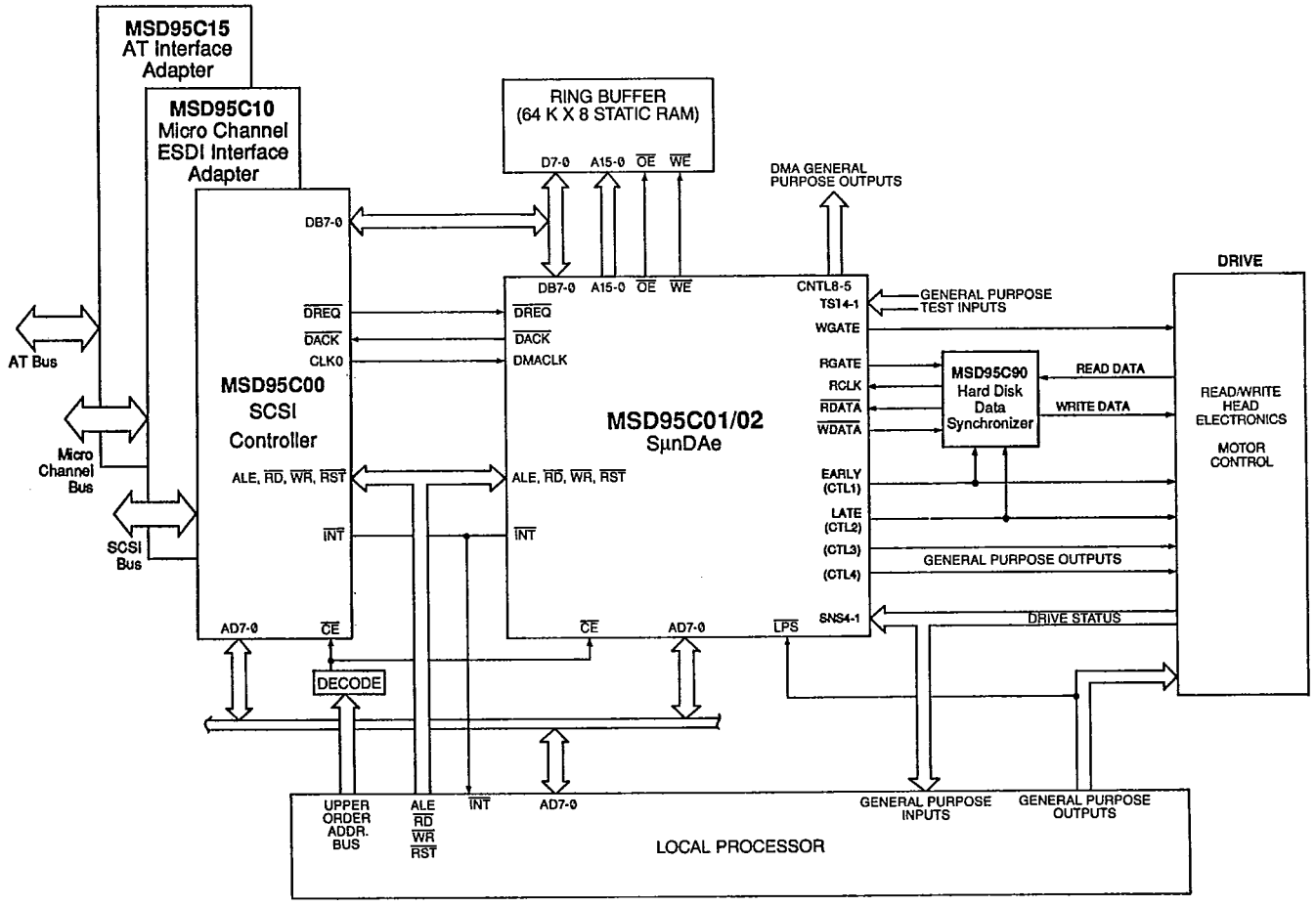


FIGURE 1: TYPICAL SYSTEM BLOCK DIAGRAM

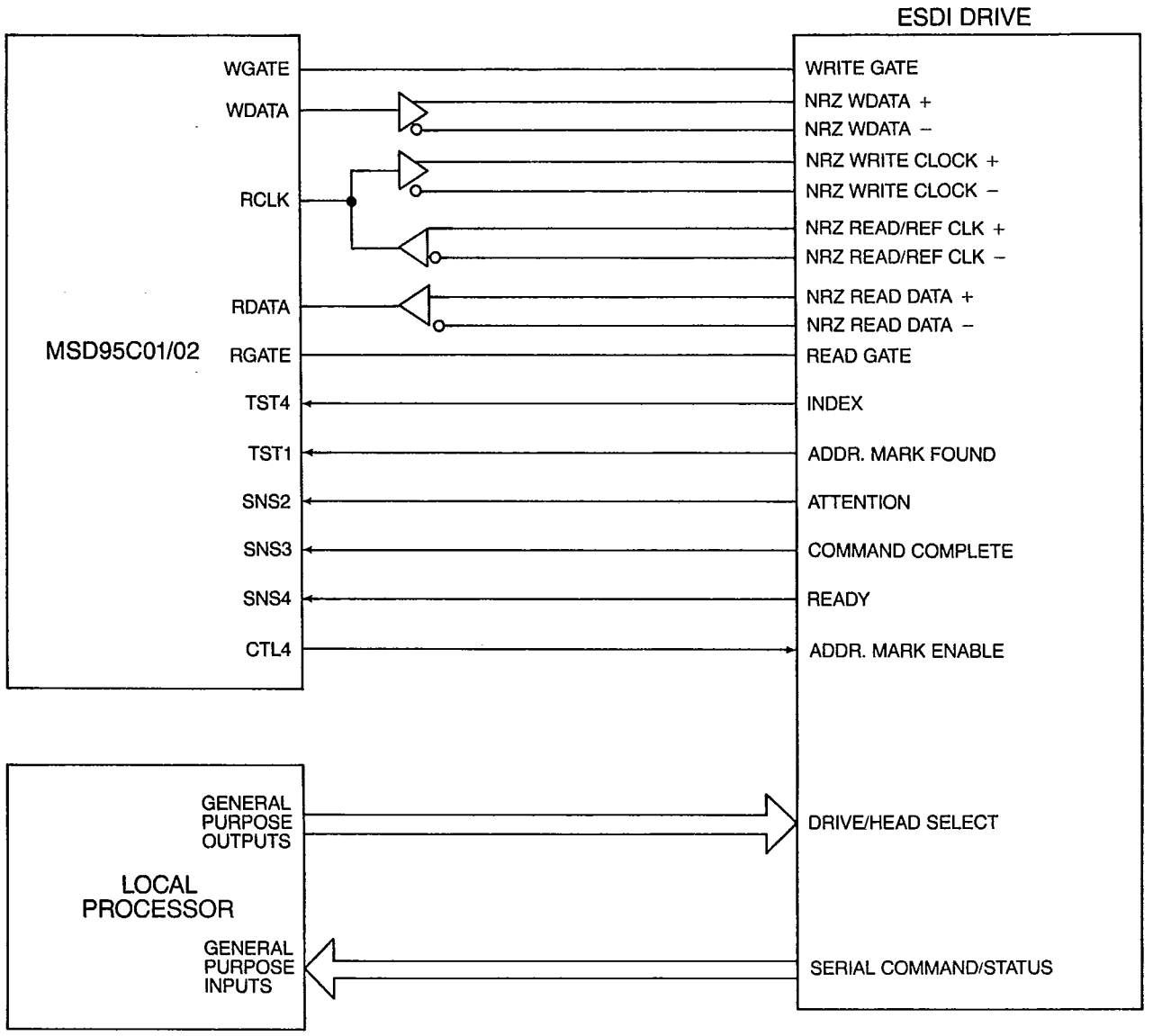


FIGURE 2: MSD95C01/02 TO ESDI INTERFACE

SYSTEM DESCRIPTION

A typical MDS95C01/02 based system consists of a LOCAL MICROPROCESSOR to control low speed tasks, a RAM BUFFER to store data to and from the disk media, a DISK INTERFACE section which may include a data separator, and a HOST BUS INTERFACE unit which connects the MSD95C01/02 to advanced system buses such as the SCSI bus, the Microchannel bus or IBM AT bus.

As can be seen in the system block diagram, the interface to the external static RAM BUFFER requires no added circuitry. A major advantage of an MSD94C01/02 based system is the use of two external data buses; one to permit data flow between the RING BUFFER and the HOST BUS INTERFACE unit (e.g. SCSI, MCA, IBM AT, etc) and the other to permit the local processor to access both the MSD95C01/02 and the HOST BUS INTERFACE unit. The two data buses allow for uninterrupted data flow in and out of the RING BUFFER during local processor updates to the DISK CONTROLLER and BUS INTERFACE devices.

LOCAL MICROPROCESSOR INTERFACE

The MSD95C01/02 is optimized to work with a microprocessor or a microcontroller having a multiplexed address/data bus without the need for any external glue logic. The main function of the local microprocessor is to interpret high level commands coming from the HOST BUS and convert them into proper instructions to the MSD95C01/02. Additionally, the local microprocessor will download micro-routines into the disk controller and handle the low speed control functions of the disk drive through the use of general purpose input and output lines.

RAM BUFFER INTERFACE

The interface to the RAM RING BUFFER is through the use of the triple DMA controller of the MSD95C01/02. The microsequencer of the MSD95C01/02 uses the RING BUFFER to store data going to and from the disk media. DMA transfers to and from the HOST bus are supported to transfer data from the RING BUFFER. Additionally, the local microprocessor has access to the RING BUFFER as well. The DMA controller section of the MSD95C01/02 controls and arbitrates up to 64K of external RING BUFFER built with standard static RAMs. The buffer size can be easily extended with the addition of a few gates. Disk data transfers to and from the buffer are managed by the MSD95C01/02 without the need for processor intervention. A DMA

Request/Acknowledge handshake is used to control HOST BUS (e.g. SCSI, MCA, etc) accesses to the RING BUFFER which are completely asynchronous with respect to disk and local processor transfers.

This method of local buffer access combined with a powerful programmable ECC (MSD95C01) or Extended Reed Solomon ECC (MSD95C02) and CRC extension designed to virtually eliminate miscorrections, provides the user with on-the-fly error correction capability with no loss of disk revolutions.

DISK INTERFACE

Interface to an ESDI type disk drive is accomplished with the use of a few driver/receiver components (see figure 2). When interfacing to an MFM or RLL type of drive, the MSD95C90 DISK INTERFACE UNIT may be used to provide data separation and pre-compensation functions. The MSD94C01/02 contain several general purpose input/output pins to control and monitor external events. In addition, there are four general purpose test inputs and four latched sense inputs that may be used directly by the microsequencer to perform conditional jumps. The sense inputs, test inputs, and the control outputs are used to efficiently handle ESDI drives.

HOST BUS INTERFACE

An assortment of HOST BUS INTERFACE circuits are provided by Standard Microsystems that interface the MSD95C01/02 to a variety of system buses. These include the MSD95C00 SCSI CONTROLLER for embedded SCSI applications, the MSD95C10 MIDAS device for Micro Channel applications and the MSD95C15 AT INTERFACE UNIT for interface to the IBM AT bus for both adapter and motherboard applications. Please refer to the appropriate data sheets for detailed information.

FUNCTIONAL DESCRIPTION

Figure 3 shows the internal block diagram of the MSD95C01/02. The data flow through the chip occurs on the INBUS and the OUTBUS. The OUTBUS connects all data coming from the ring buffer to the drive and the INBUS connects all data from the drive to the ring buffer. The blocks which make up the MSD95C01/02 are described as follows:

DMA CONTROLLER

As shown in Figure 4, the three channel DMA controller is composed of a 6 X 16 bit register file, two OFFSET COUNTERS for monitoring ring buffer full/empty status, a DMA FUNCTION Register to indirectly address the register file, a ring buffer DATA Register, and ALU to perform incrementing (by 1, 2, 3 or 4) and in addition, a 16 bit mailbox register, state controller and DMA request priority resolver.

Channel access to the ring buffer comes from three sources. The MICROSEQUENCER requests are initiated by the DISK INTERFACE block for every byte transferred to/from the MICROSEQUENCER and is given top priority. Local processor accesses to the ring buffer are initiated by direct operation of the local processor on the addressable registers within the DMA controller. The local processor is given second priority. External Device access (like SCSI) is initiated via a DMA request-acknowledge handshake and is given third priority.

ALU and Register function:

For each DMA channel, specific register manipulations are possible depending upon the operation required. Each channel will have different requirements on the way the address to the ring buffer is updated. For instance, during disk read operations, the channel must ensure that the starting address for a sector transfer is not updated until the data transferred into the ring buffer is known to be error free. The External Device, during a read operation from the ring buffer, must cause the address to be incremented for each byte transferred. It must also update the OFFSET COUNTER since there is a relationship between the value in the OFFSET COUNTER and the amount of available free space there is in the buffer and available error free data there is to transfer out on the External channel. The local processor channel might be required to perform read modify write routines during error correction operations.

Local Processor channel operation:

The local Processor can specify the address operation to be performed during its channel access via the DMA FUNCTION Register which is a write only local processor addressed register (assigned address 50HEX). This function register permits the loading of all registers in the register file which are not directly addressable from the external local processor.

The Local Processor can directly access four registers in the DMA controller. They are:

DMA FUNCTION Register (write only) address 50H
MAILBOX HIGH Register (read/write) address 51H
MAILBOX LOW Register (read/write) address 52H
DATA Register (read/write) address 53H

The local processor can indirectly access 6 additional 16 bit registers that make up the DMA register file. These registers are accessed by using the DMA FUNCTION Register. An example of their use is illustrated as follows:

CONSTANT 1—Normally used to hold the sector size and in calculating address updates and ring buffer status flags.

CONSTANT 2—Normally used to hold a number that is a function of the Ring Buffer size for generating Ring Buffer empty/full status.

LOCAL PROCESSOR ADDRESS Register—Holds the current address of a local processor to ring buffer data transfer.

EXTERNAL DEVICE ADDRESS Register—Holds the current address of an External Device to ring buffer data transfer.

DISK Register—Maintains the address of the first location of a MICROSEQUENCER initiated disk transfer until data integrity is established.

DISK ADDRESS Register—Holds the current address of a MICROSEQUENCER initiated disk ring buffer data transfer.

DMA FUNCTION Register–Refer to Table 9 for details.

Disk Channel Access Operation:

Disk requests for data transfer are initiated by the MSD95C01/02's internal MICROSEQUENCER. For normal data transfer between the disk and the ring buffer, the MICROSEQUENCER causes the DISK ADDRESS in the DMA block to be incremented by one between ring buffer access cycles.

In addition, the MICROSEQUENCER may perform some housekeeping functions at each sector boundary if required. These housekeeping functions may include updating the OFFSET COUNTER, determining buffer full/empty status, etc.

In order to specify one of many housekeeping calculations, the MSD95C01/02's internal MICROSEQUENCER has the ability to specify the operation by loading a DISK DMA FUNCTION Register (different from the local processor addressable DMA FUNCTION Register). The bit definitions

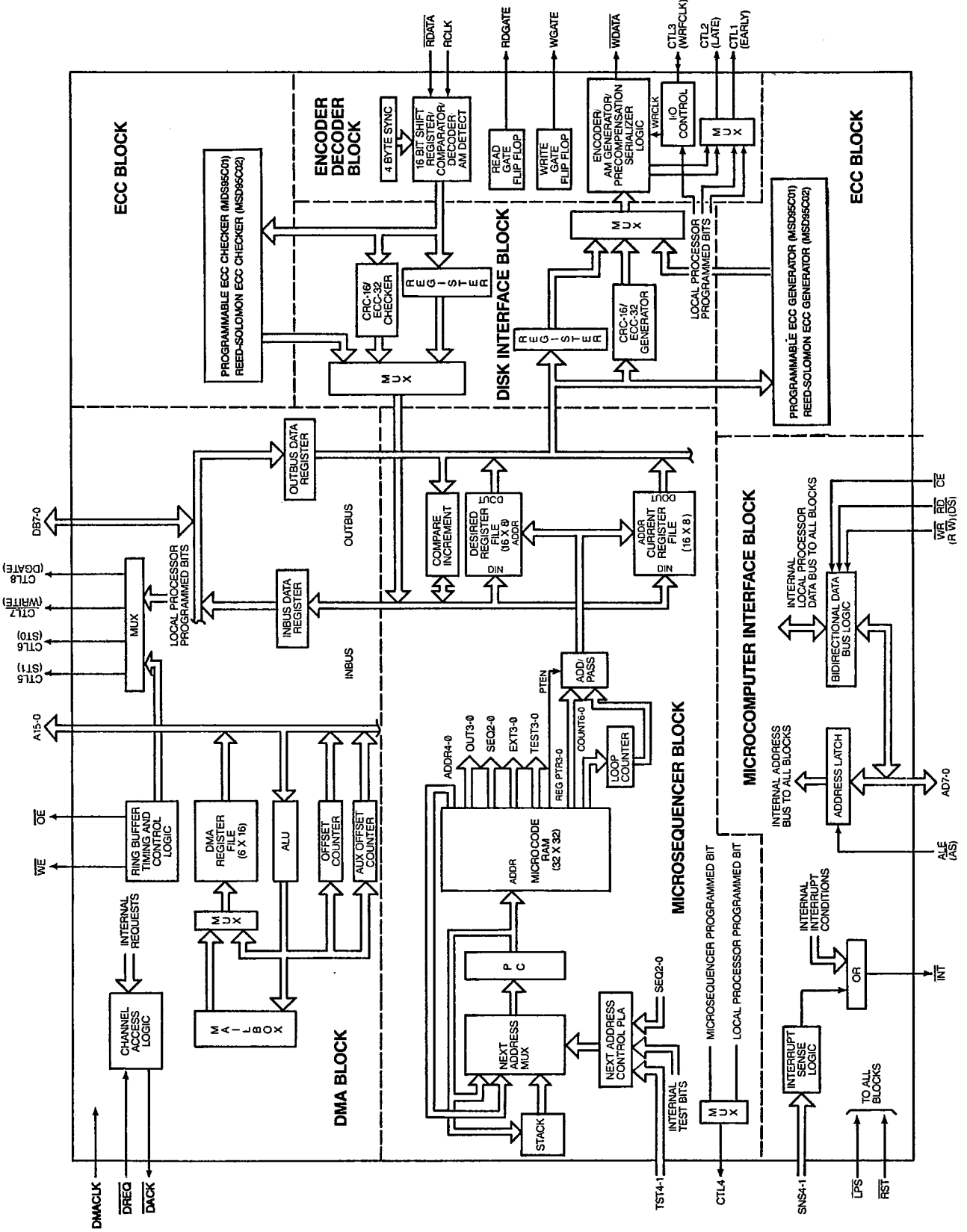


FIGURE 3: MSD95C01/02 INTERNAL BLOCK DIAGRAM

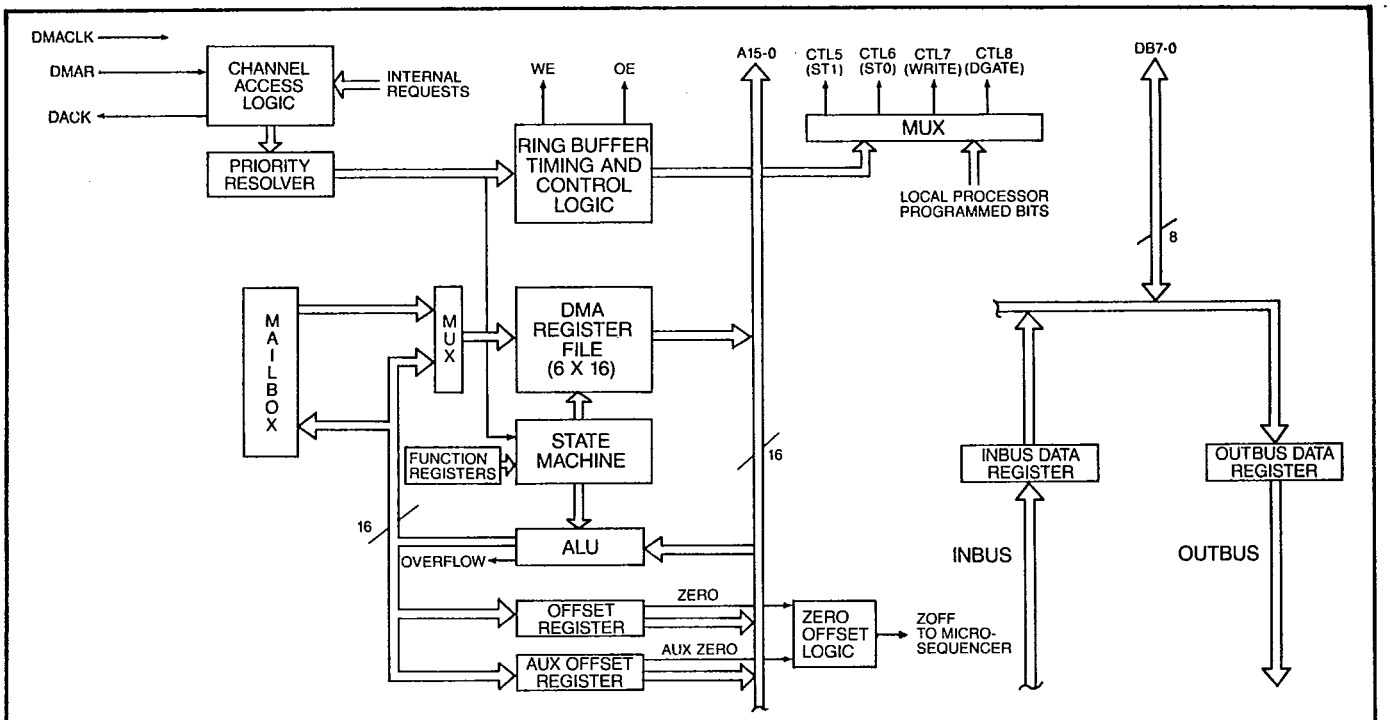


FIGURE 4: DMA BLOCK DIAGRAM

of this register are identical to the DMA FUNCTION Register as defined in Tables 1 and 2. The MICROSEQUENCER can load the DISK DMA FUNCTION Register by transferring a value previously stored in the DESIRED REGISTER FILE to the DMA block.

EXTERNAL DEVICE CHANNEL ACCESS OPERATION:

Upon an external DMA request, this channel will perform one hardwired function consisting of proper adjustment of the current ring buffer address as well as decrementing the OFFSET COUNTERs for current handling of ring buffer full/empty situations. The MSD95C01/02 can buffer a maximum of 12 DMA requests before issuing an acknowledgment without causing an overrun/underrun condition. The MSD95C01/02 uses the leading edge of DMA Request to post the EXTERNAL channel access. This permits the MSD95C01/02 to work with several REQ-ACK timing situations.

For External Device DMA operations, the DMA hardwired function performed is:

EXTERNAL DEVICE ADDRESS + N → EXTERNAL DEVICE ADDRESS.

N is used to permit various data bus widths between the ring buffer and the External Device as shown in table 1.

N	DATA BUS WIDTH
1	8 BITS
2	16 BITS
3	24 BITS
4	32 BITS

TABLE 1: EXTERNAL DEVICE BUS WIDTHS AS A FUNCTION OF N

The value N is programmed by the Local Processor in MODE 1 Register, bits 5 and 4.

OFFSET COUNTER:

The OFFSET COUNTER is part of the automatic housekeeping function of the DMA block and is used to keep track of buffer empty/full conditions.

During External Device to Ring Buffer to disk operations (WRITE the disk), the OFFSET COUNTER keeps track of the number of free bytes left in the buffer. Whenever the Ring Buffer size minus the OFFSET COUNTER is less than the sector size (indicating that there is less than a sector's worth of data in the ring buffer), the disk WRITE is temporarily held off. The throttling of the Disk DMA channel as a function of buffer space status is done automatically by the DMA controller without local processor intervention.

During Disk to Ring Buffer to External Device (READ the disk) operations, the OFFSET COUNTER keeps track of the total data bytes left in the buffer. Whenever the Ring Buffer size minus the OFFSET COUNTER is less than the sector size (indicating that there is not enough room in the buffer to accept another sector), the disk READ is temporarily held off. The throttling of the Disk DMA channel as a function of buffer space status is done automatically by the DMA controller without local processor intervention.

Decisions by the MICROSEQUENCER regarding buffer empty/full status are made by interrogating the ZOFF flag whenever an ALU operation loads a zero into the OFFSET COUNTER.

AUXILIARY OFFSET COUNTER operation:

The AUXILIARY OFFSET COUNTER is loaded and modified along with the OFFSET COUNTER. It is required because data transferred from the disk into the Ring Buffer might contain errors. Because of this, the AUXILIARY OFFSET COUNTER might be different from the OFFSET COUNTER and is used to control the flow of data between the External Device and the Ring Buffer. During External Device to Ring Buffer transfers, the AUXILIARY OFFSET COUNTER keeps track of the number of free bytes left in the buffer. During Ring Buffer to External Device transfers, the AUXILIARY OFFSET COUNTER keeps track of the number of error free data bytes left in the buffer.

The OFFSET and AUXILIARY OFFSET COUNTERS are linked together until an error is detected on data read from the disk. When this occurs, only the OFFSET COUNTER is incremented as new data is transferred from the disk to the Ring Buffer. The AUXILIARY OFFSET COUNTER is not incremented until the error is corrected. Both counters are decremented when data is transferred from the Ring Buffer to the External Device. If the error is corrected before the AUXILIARY OFFSET COUNTER is decremented to zero, then the two counters are linked back together by transferring the contents of the OFFSET COUNTER into the AUXILIARY OFFSET COUNTER in response to a DMA FUNCTION Register command. If the AUXILIARY OFFSET COUNTER is decremented to zero before the error is corrected, the MSD95C01/02 will not respond to DMA requests from the external device until the AUXILIARY OFFSET COUNTER and OFFSET COUNTER are linked back together. See Appendix 4 for a further description of the OFFSET and AUXILIARY OFFSET COUNTERs during error correction on the fly.

Status Flags:

The DMA block will generate three status flags which can be used by the MICROSEQUENCER to test and make decisions on the microprogram flow. These three status flags, defined as follows, are also readable by the Local Processor in STATUS 1 Register (ADDR 55H):

OVERFLOW: During a DMA housekeeping cycle, this flag indicates the result of the current operation performed by the ALU; if the arithmetic yields an overflow, this bit will be set. This flag is latched during a DMA housekeeping cycle and will continue to indicate the result of the operation until the next housekeeping cycle latches a new OVERFLOW status. This bit is available to the Local Processor in STATUS 1 Register, bit D5.

ZERO: This flag is set to a one when the OFFSET COUNTER is decremented to zero. This bit is available to the Local Processor in STATUS 1 Register, bit D6.

AUX ZERO: This flag is set to a one when the AUXILIARY OFFSET COUNTER is decremented to zero. This bit is available to the Local Processor in STATUS 1 Register, bit D7.

The ZERO and AUX ZERO flags are not set if the OFFSET and AUXILIARY OFFSET COUNTERS are loaded with a zero.

The MICROSEQUENCER can interrogate the logical OR of the ZERO and AUX ZERO flags via the test input ZOFF.

ECC ON THE FLY:

The three channel DMA arrangement provides the user with the ability to perform error correction on the fly without loss of a disk revolution. In general, upon disk read operations, one sector may be transferred from the Ring Buffer over the EXTERNAL channel, one sector may be operated on by the Local processor for error correction, and the third sector may be read from the disk and written into the Ring Buffer. Refer to Appendix 4 for further description of ECC on the fly.

MICROSEQUENCER

As shown in Figure 5, the MICROSEQUENCER consists of a 32 X 32 microcode RAM, a 7 bit loop counter, a one address STACK, a sophisticated next address generator and two 16 byte register files. During next address generation, the Program Counter can be loaded from the STACK, from the ADDRESS FIELD output by the microcode RAM (ADDR 4-0), or from the current or incremented value in the Program Counter.

The Local Processor can initiate Command execution (eg. a Read Command) by writing to the COMMAND START Register (ADDR 54 HEX). The MSD95C01/02 will then begin execution at address zero in the microcode RAM. From this time until the command terminates, program flow is dependent on which of the several sources are specified when the Program Counter is loaded. Selection of the next address is dependent on the Sequence control field (SEQ 2-0), and internal and external test points which are input to the Next Address Control PLA. Test points are chosen via the Test field (TEST 3-0) for interrogation and program flow of the MICROSEQUENCER.

Contained within the MICROSEQUENCER block are two, 16 byte register files named DESIRED and CURRENT REGISTER FILEs. In general, the MICROSEQUENCER reads the DESIRED REGISTER FILE and writes the CURRENT REGISTER FILE. The Local Processor can read or write either register file. Local Processor access to these register files are restricted and controlled by interrupts generated by the MICROSEQUENCER to the Local Processor. This limited access is required to resolve the access contentions to these register files by both the Local Processor and the MICROSEQUENCER.

In addition to the DESIRED and CURRENT REGISTER FILEs, there is a comparator structure set up to compare contents in the two register files as shown in Figure 8. Any information loaded by the local processor in the DESIRED REGISTER FILE may be compared with the data that is deposited in the CURRENT REGISTER FILE as it is filled with data read from the disk. The sequence and number of compares made are all initiated under MICROSEQUENCER control. Typically, the compares can be performed to determine whether a particular ID field had been encountered.

The contents of any DESIRED REGISTER FILE location can be incremented under MICROSEQUENCER control. Typically, this can be used to increment the sector number in the DESIRED REGISTER FILE location when performing consecutive logical sector operations.

In addition, the CURRENT REGISTER FILE can, under MICROSEQUENCER control, be loaded with the error syndrome bytes for examination by the Local Processor during an error correction operation.

In typical applications, the CURRENT REGISTER FILE holds data such as the current header information (head #, sector #, track #) and the writing to these registers is controlled by the MICROSEQUENCER as data is converted from serial to parallel in the DISK INTERFACE block. The DESIRED REGISTER FILE holds information such as the desired sector to be operated on in a READ or WRITE DISK operation. The desired header information, in this case is written into the DESIRED REGISTER FILE by the Local Processor prior to the execution of the READ OR WRITE command in an order that is consistent with the order in which the MICROSEQUENCER loads and compares the CURRENT REGISTER FILE.

The 32 bit microprogram word definition contains a total of 8 separate fields as shown in Figure 6.

Description of Individual Fields:

Sequence Control Field (D31-29):

The SEQUENCE CONTROL FIELD determines the next address loaded into the PROGRAM COUNTER which feeds the MICROSEQUENCER RAM. Depending upon which SEQUENCE CONTROL FIELD is specified, the next address might remain at the current address, increment, use the value stored in the STACK, or use the value specified in

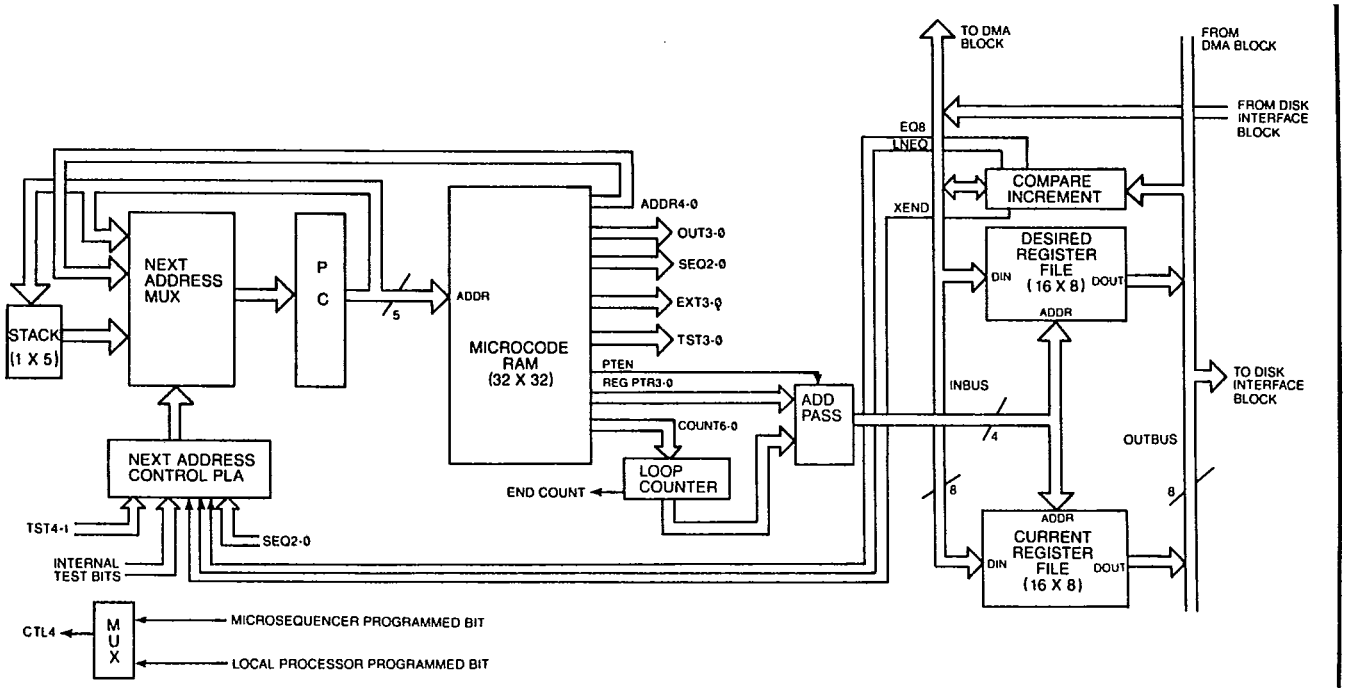


FIGURE 5: MICROSEQUENCER BLOCK DIAGRAM

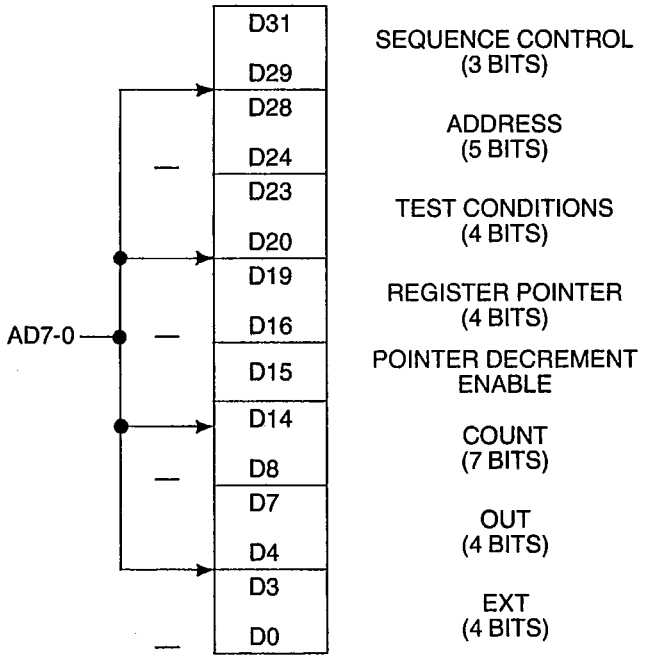


FIGURE 6: 32 BIT MICROSEQUENCER PROGRAM WORD DESCRIPTION

the ADDRESS FIELD. In addition, it can cause the current value in the PROGRAM COUNTER to be pushed on the STACK. The STACK is one address deep. The END COUNT and TEST POINT can affect the next address generation and STACK operations in a number of ways. See section on the TEST FIELD and COUNT FIELD for a description of the various conditions that will produce a valid TEST POINT or END COUNT.

Address Field (D28-24):

The Address Field bits in the microcode instruction may be directly loaded into the Program counter when executing "jump" and "call" instructions. When executing a call instruction, the current (or next) Program Counter value may be saved on a one address STACK. "Return" instructions can be implemented by loading the contents of the STACK back into the Program Counter. It should be noted that subroutines can only be nested one deep.

Test Conditions Field (D23-20):

The test condition field permits the MICROSEQUENCER to test one of 16 conditions for the purpose of conditional jumps and calls as specified in the SEQUENCE CONTROL FIELD. These 16 inputs originate in the MICROSEQUENCER block and other blocks and four of the test points are general purpose inputs from outside the MSD95C01/02 appearing on pins TST1-4.

T-52-33-63

SEQUENCE CONTROL FIELD DEFINITION:

MNEMONIC	D31-29	OPERATION	TEST POINT END COUNT	1 1	1 0	0 1	0 0
SHLP (SHORT LOOP)	000	PC = STACK =		PC+1 D28-24	PC —	PC+1 —	PC —
LGLP (LONG LOOP)	001	PC = STACK =		PC+1 D28-24	PC —	PC+1 —	PC —
RC (RETURN OR CALL)	010	PC = STACK =		STACK —	PC —	D28-24 PC+1	PC —
CI (CALL OR INCREMENT)	011	PC = STACK =		D28-24 PC+1	PC —	PC+1 —	PC —
TSJ (TEST JUMP)	100	PC = STACK =		PC+1 —	PC+1 —	D28-24 —	PC —
TSC (TEST CALL)	101	PC = STACK =		PC+1 —	PC+1 —	D28-24 PC+1	PC —
TSJL (TEST JUMP LONG)	110	PC = STACK =		PC+1 —	PC+1 —	D28-24 —	PC —
TSCL (TEST CALL LONG)	111	PC = STACK =		PC+1 —	PC+1 —	D28-24 PC+1	PC —

1. D28-24 is the address as specifird in the ADDR field of the microcode word.
2. — = no change.

TEST FIELD DEFINITION:

D23	D22	D21	D20	TEST POINT SELECTED	DEFINITION
0	0	0	0	FORCE ZERO	Forces the TEST input to a logic zero.
0	0	0	1	DC GAP	From the ENCODER/DECODER block indicating that no transitions have occurred on the RDATA input pin for 15 clocks.
0	0	1	0	SYNC	From the DISK INTERFACE block indicating a 16 bit high speed compare between disk data and the MICROSEQUENCER selected SYNC Register(s). This signal is used to start all CRC and ECC generation logic.
0	0	1	1	INTER*	From the DISK INTERFACE block indicating that a CRC-16, ECC-32, Programmable ECC (MSD95C01) or Reed Solomon ECC (MSD95C02) data error has occurred. In tape read after write mode, a one indicates a data error. Otherwise, a zero indicates a data error.
0 0 0 0	1 1 1 1	0 0 1 1	0 1 0 1	TST1 TST2 TST3 TST4	Input pin TST1. Input pin TST2. Input pin TST3. Input pin TST4.
1	0	0	0	EQ	This input reflects the status of an 8 bit compare between the INBUS and the DESIRED REGISTER FILE.
1	0	0	1	LNEQ	Indicates that one of the previous 8 bit compares or the current 8 bit compare is not equal.
1	0	1	0	XEND	Indicates that the contents of a DESIRED REGISTER FILE location has been incremented to all zeros.
1	0	1	1	NTP2	This is an auxiliary test input which is the logical OR of four independent conditions. It is equal to either a HALT (local processor initiated via Register 54, bit 0), an ABNORMAL DATA MARK (Register 58H, bit 6), a WRITE FAULT (Register 58H, bit 5), or, in tape mode, if the sync was not found in read after write. A zero indicates that one of the conditions is true.
1	1	0	0	CMD	Originates in the DMA block and indicates if the command loaded by the local processor is a READ or WRITE Command with respect to the Disk. This operation tests bit 6 inverted in the START COMMAND Register. A one is a disk write and a zero is a disk read.
1	1	0	1	BFRDY	Originates from the DMA block. This test input is a logic one if the result of an ALU operation has not yielded an overflow and NTP2 is a logic one.
1	1	1	0	ZOFF	Originates from the DMA block and indicates when either the OFFSET or AUXILIARY OFFSET COUNTER is loaded with a zero as a result of an ALU operation.
1	1	1	1	ONE	Forces the TEST input to a logic ONE.

Register Pointer Field (D19-16):
The REGISTER POINTER FIELD is associated with the DESIRED and CURRENT REGISTER FILES and specifies the addess of the particular register file location to be operated on. The DESIRED and CURRENT REGISTER FILEs have a common address such that the same corresponding location in each file may be operated on at the same time.

Pointer Decrement Field (D15):
This one bit field permits one to access sequentially decrementing locations in the DESIRED and CURRENT REGISTER FILEs for multiple operations within these registers.

If the Pointer Decrement enable bit is zero, the Register Address will be the value specified in the Register Pointer field. If the Pointer Decrement enable bit is one, the Register Address will be the value specified in the Register Pointer field added to the current contents of the Loop Counter. When accessing sequential register file locations, the Count field should be initialized to the number of sequential access required and the REGISTER POINTER FIELD should be initialized to the last sequential register file location to be accessed. At each MICROSEQUENCER clock, the loop counter in the instruction is decremented, thus permitting sequential access at the MICROSEQUENCER clock frequency.

Count Field (D14-8):

This field serves two functions. First, it is loaded into the loop counter for counting microinstruction loops. In this instance, the generation of an END COUNT status bit, will cause branching conditions as defined by the SEQUENCE CONTROL FIELD (D31-29). The END COUNT status bit will be set to a logic one when the loop counter is decremented to zero. The second function of this field is for accessing sequential locations of the DESIRED and CURRENT REGISTER FILEs as described in the POINTER DECREMENT ENABLE FIELD.

It should be noted that the count value is loaded into the loop counter as the microinstruction containing it is exited.

The loop counter is decremented every microinstruction clock or every 128th microinstruction clock for a short loop (SHLP) or a long loop (LGLP) respectively. To perform a count of N, the value N-1 should be specified in the previous instruction in the microcode since the loop counter gets loaded as an instruction is exited.

Out Field (D7-4):

The out field is one of two control fields used to control internal and external operations during microprogram execution. This four bit field yields one of 16 control outputs as follows:

OUT FIELD DEFINITION:

MNEMONIC	D7-D4	OPERATION
NOP	0 0 0 0	No operation.
AMEN	0 0 0 1	To DISK INTERFACE block causing the output CTL 4 to change. This output is multiplexed with a programmable output controlled by the local processor. This signal may be used for an external address mark enable when interfacing with ESDI drives.
RDG	0 0 1 0	To the DMA block. This signal is a DMA request from the MICROSEQUENCER and enables disk data to get transferred to the DATA Register in the DMA block over the INBUS. It also can cause the output DG (Data Gate) to go active. The DG output is multiplexed with a programmable output controlled by the local processor.
FDMA	0 0 1 1	This signal turns DGATE off.
WPRE	0 1 0 0	This signal allows the PLO SYNC (preamble) data stored in the DESIRED REGISTER FILE to be loaded into the ENCODER/DECODER/DISK INTERFACE block over the OUTBUS. In addition, during a write, it is used to preset all CRC and ECC generation logic.
WMISS	0 1 0 1	*For disk drive application:* This signal allows the missing clock data pattern stored in the DESIRED REGISTER FILE to be loaded into the ENCODER/DECODER/DISK INTERFACE block over the OUTBUS and generate the missing clock pattern when shifting the data out to the disk. *For tape application:* This signal is used to write special marks in GCR code. To write out a block of "file mark" data (00101 00101), write FFH data with WMISS in out field. To write out a byte of "Data Mark" data (11111 00111), write out 5DH data with WMISS in out field. In addition, during a write, WMISS is used to start all CRC and ECC generation logic and to activate CTL4.
FREG	0 1 1 0	This signal gates the data stored in the DESIRED REGISTER FILE on to the OUTBUS.
SNDRM	0 1 1 1	This signal gates the error syndrome (from the selected error checker) to the CURRENT REGISTER FILE.
PREQ1	1 0 0 0	This signal causes the contents of the DESIRED REGISTER FILE pointed to by the register pointer field to be loaded into the DISK DMA FUNCTION Register.
WCRC	1 0 0 1	This signal gates the check bytes (CRC-16, ECC-32, Programmable ECC (MSD95C01), or Reed Solomon ECC (MSD95C02) out to the disk.
DELDM	1 0 1 0	This signal enables the deleted data mark.
	1 0 1 1	RESERVED FOR FUTURE USE.
PINT0	1 1 0 0	This signal sets bit 5 of STATUS 2 Register (ADDR 56H) and will, if enabled, generate an interrupt to the Local Processor. In addition, this interrupt will permit the Local Processor to access the CURRENT REGISTER FILE. It also separates the OFFSET and the AUXILIARY OFFSET COUNTERS. Typically, this interrupt is used for error correction operations. See description of STATUS 2 Register.

OUT FIELD DEFINITION (CONTINUED):

MNEMONIC	D7-D4	OPERATION
PINT1	1 1 0 1	This signal sets bit 6 of STATUS 2 Register (ADDR 56H) and will, if enabled, generate an interrupt to the Local Processor. In addition, this interrupt will permit the Local Processor to access the CURRENT REGISTER FILE. Typically, this interrupt may be used to permit the Local Processor to read the current ID information that is stored in the CURRENT REGISTER FILE.
PINT2	1 1 1 0	This signal sets bit 7 of STATUS 2 Register (ADDR 56H) and will, if enabled, generate an interrupt to the Local Processor. This interrupt is general purpose and can be generated upon recognition of any MICROSEQUENCER detectable condition.
DNINT	1 1 1 1	This signal sets bit 4 of the INTERRUPT STATUS Register (ADDR 54H) and will, if enabled, generate an interrupt to the Local Processor. Typically, this signal is invoked to inform the Local Processor that the current command executed by the MICROSEQUENCER has been completed. Generation of this signal will stop the MICROSEQUENCER, reset BUSY (STATUS 3 Register, Bit D7), and set done (INTERRUPT STATUS Register, Bit D4). In addition, this interrupt will permit the Local Processor to access the CURRENT REGISTER FILE.

EXT Field (D3-0):

The EXT field is one of two control fields used to control internal and external operations during microprogram execution. This four bit field yields one of 16 control outputs as follows:

EXT FIELD DEFINITION:

MNEMONIC	D3-D0	OPERATION
NOOP	0 0 0 0	No operation.
SWDG	0 0 0 1	This signal posts a DMA request for the MICROSEQUENCER to the DMA block and is used during disk write operations. The signal allows data to be transferred from the DATA Register in the DMA block to the DISK INTERFACE and ENCODER/DECODER blocks. It also generates the DG (Data Gate) signal.
SRCRC	0 0 1 0	This signal is used to inform the CRC-16 or ECC-32 CHECKER that the respective code is being transferred from the disk.
SRECC	0 0 1 1	This signal is used to inform the Programmable ECC CHECKER (MSD95SC01) or the Reed Solomon ECC CHECKER (MSD95SC02) that the ECC bytes are being transferred in from the disk.
SGAP	0 1 0 0	RESERVED FOR FUTURE USE.
SPRE	0 1 0 1	This signal enables the compare between incoming data and the value programmed in the appropriate SYNC register(s). Normally, this comparison is used to detect a PLO SYNC (preamble) data pattern. The RGATE signal is also set. This signal presets all CRC and ECC generation logic during a read.
SAM	0 1 1 0	This signal enables the compare between incoming data and the value programmed in the appropriate SYNC register(s). Normally, this comparison is used to detect a missing clock data pattern. The RGATE signal is also set. During a read, when SYNC is detected and this signal is active, all CRC and ECC generation logic is started.
SDM	0 1 1 1	This signal enables the compare between incoming data and the value programmed in the appropriate SYNC register(s). Normally, this comparison is used to detect a data mark data pattern. The RGATE signal is also set.
SGOFF (1)	1 1 0 0	This signal resets both RGATE and WGATE.

NOTE: The signals above describe operations to be performed as the instruction containing them is exited (The PC is changed). The following signals will be invoked when the instruction containing them is entered.

MNEMONIC	D3-D0	OPERATION
TREG	1 0 0 0	This signal gates the contents of the INBUS into the CURRENT REGISTER FILE as pointed to by the REGISTER POINTER FIELD.
TDMA	1 0 0 1	This signal enables disk data to get transferred to the DATA register in the DMA block over the INBUS.
SUB	1 0 1 0	Enables CRC16 or CRC32, as designated in the Mode 4 Register bit D2.
MAIN	1 0 1 1	Enables CRC16, CRC32, Programmable ECC (MSD95C01) or Reed Solomon ECC (MSD95C02), as designated in the Mode 4 Register, bits D5-D3.
ZRCLR	1 1 0 1	This signal clears the ZERO and AUX ZERO flags in the DMA block.
INCE	1 1 1 0	This signal causes the contents of the DESIRED REGISTER FILE pointed to by the register pointer field to be incremented and placed back into the same DESIRED REGISTER location.
WGON	1 1 1 1	This signal sets WGATE on.

1) The signal SGOFF is shown out of numerical sequence to permit it to be grouped accordingly.

DISK INTERFACE & ENCODER/DECODER BLOCK

This block interfaces to the rest of the chip via the INBUS and the OUTBUS which connects the serial to parallel/parallel to serial converters to the DESIRED and CURRENT REGISTER FILES and the appropriate data registers within the DMA block.

In addition to serialization and deserialization of data, this block will perform complex high speed compare logic that can sequentially compare a bit pattern read from the disk of up to 32 bits in length. Any and all compares can detect a predefined missing clock pattern. The sequence of comparisons made prior to declaring a valid synchronization with the rotating media, is fully programmable under MICROSEQUENCER control. During write disk operations, this block can generate fully programmable formats and complex missing clock data patterns completely under MICROSEQUENCER control.

The encoder/decoder section is local processor programmable to handle FM, MFM, RLL 2, 7 and GCR codes. Included in this block is the CRC-16 and IBM AT compatible 32 bit ECC generator and checker.

Programmable Polynomial Error Correction Code (ECC) (Exists Only In MSD95C01)

This ECC block contains a fully programmable polynomial generator. The user can specify the degree, from one to 64 in one bit increments, and feed back terms to satisfy the required error burst correction span, pattern sensitivities, miscorrection probability, and maximum record length coverage. Cyclical redundancy codes (CRC), or polynomials used exclusively for detecting errors when correction information is not necessary, can also be implemented with the programmable ECC block. When programming a polynomial of length one, a parity check in the incoming data stream is obtained.

The ECC block premultiplies the incoming data stream by the degree programmed in the size register and concurrently divides by the polynomial programmed in the feed back tap registers. If an error occurs on a read operation, an error syndrome is available yielding location and error value information for software correction procedures. Programmability also provides the option of setting the polynomial to all zeros or all ones and including sync and data marks in ECC calculation.

Extended Reed Solomon Error Correction Block (Exists Only In MSD95C02)

The EXTENDED REED SOLOMON ECC block has been optimized to permit single and double burst error correction while simultaneously reducing the probability of miscorrection and expanding the error detection capability dramatically.

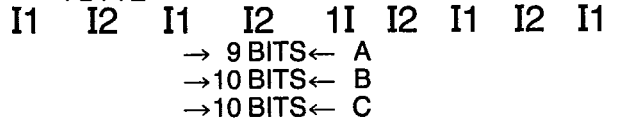
Fundamental of Reed-Solomon Error Correcting Codes

The error correction and detection capability of any Reed Solomon code always relates the number of symbols (bytes) that are allowed to be in error to permit valid correction and detection. In the simplest case, data is run through hardware one byte at a time and, depending on the complexity of the hardware and the number of redundancy bytes (bytes appended to the data field) used, one can allow a certain number of these bytes to be in error and still recover the data via correction and or detection.

To minimize hardware and maximize the error detection and correction capability of the Reed Solomon code, several identical hardware constructions are employed in parallel; each with their individual limit on the number of symbols (bytes) that can be in error. This method is called interleaving. The interleaving method is set up such that the first byte goes into the first interleave, the second byte into the second interleave, etc. for the entire length of the data plus redundancy fields. The MSD95C02 can employ either two or three interleaves, each interleave containing a maximum of 255 bytes. Further, each interleave can be programmed to handle either 1 or 2 symbols in error. The combination of interleaves and allowable symbols in error per interleave, accounts for a variety of local processor programmable error detection and correction capabilities. A few examples will illustrate.

Case 1:

Interleave of 2 and the ability to correct 1 byte per interleave



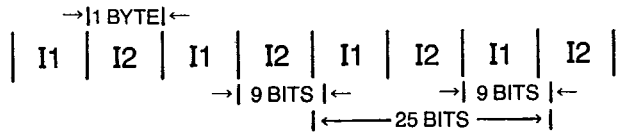
A: Any and all 9 bit bursts will never span more than one symbol per interleave. This arrangement permits correction of any single 9 bit burst error.

B: This 10 bit burst includes two symbols from interleave 1 and hence, cannot guarantee proper correction.

C: This 10 bit burst, because of its position, does not include more than one symbol per interleave and can be corrected properly.

Case 2:

Interleave of 2 and the ability to correct 2 byte per interleave.



A: Any and all 9 bit bursts will never span more than one symbol per interleave. This arrangement permits correction of any single 9 bit burst error.

B: Any single burst error, 25 bits long or smaller, will never include more than two symbols per interleave. This arrangement permits correction of any single 25 bit burst error.

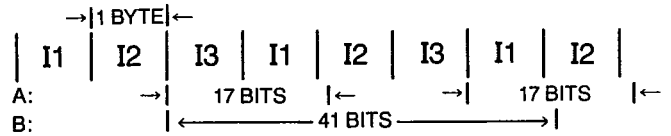
Case 3:

Interleave of 3 and the ability to correct 1 symbol per interleave.

A: Any and all 17 bit bursts will never include more than one symbol per interleave. This arrangements permits correction of any single 17 bit burst error.

Case 4:

Interleave of 3 and the ability to correct 2 symbols per interleave.



A: Any two randomly placed 17 bit bursts will never include more than two symbols per interleave. This arrangement permits correction of any two, 17 bit burst errors.

B: Any single burst error, 41 bits long or smaller, will never include more than two symbols per interleave. This arrangement permits correction of any single 41 bit burst error.

The MSD 95C02 has the ability to handle 2 or 3 interleaves and can be programmed to correct either single or double burst errors of varying length. Single or double burst error correction capability corresponds to 1 or 2 symbol errors per interleave.

The MSD 95C02 provides the user with an additional option used to control the range of the error detection capability. This is performed by including either an ECC extension field or a CRC addition field.

CRC Addition Field:

A CRC Addition Field can be appended to the data block before the Reed Solomon ECC redundancy bytes to provide greater error detection capability. The unique two or three byte CRC sequence is not interleaved and provides the ability to detect miscorrections when error bursts are beyond the Reed Solomon correction capabilities. This requirement is suggested when performing "correction on the fly" without disk revolution. The CRC Addition Field consists of two bytes (for an interleave of 2) or three bytes (for an interleave of 3) and is included in the Reed Solomon redundancy calculation. The redundancy order is as follows:

DATA FIELD	CRC1	CRC2	CRC3	ECC1, 2,...,N

ECC EXTENSION FIELD:

Instead of the CRC Addition Field, an additional 2 or 3 byte Extension Field may be appended behind the Reed Solomon bytes. This Extension Field guarantees the detection of one extra symbol in error for each interleave in the processed data block. The Reed Solomon redundancy bytes are included in the extension calculation and the number of extension bytes is equal to the selected interleave. The redundancy order is as follows:

DATA FIELD	ECC 1,2,...,N	EXT1	EXT2	EXT3

The number of redundancy bytes appearing at the end of the data field is a function of the interleave, the number of burst error correction selected, and the inclusion of the CRC Addition Field or the ECC Extension Field.

CRC Addition:

INTER-LEAVES	TOTAL REDUNDANCY BYTES	CORR TYPE	MAXIMUM DATA BLOCK LENGTH
2	6 (= 4 R/S ECC + 2 CRC)	SEC, MED	504 BYTES
2	10 (= 8 R/S ECC + 2 CRC)	DEC, MED	500 BYTES
3	9 (= 6 R/S ECC + 3 CRC)	SEC, MED	756 BYTES
3	15 (= 12 R/S ECC + 3 CRC)	DEC, MED	750 BYTES

SEC = single error correction.
DEC = double error correction.
MED = multiple error detection as defined by the CRC extension used.

ECC Extension: T-52-33-63

INTER-LEAVES	TOTAL REDUNDANCY BYTES	CORR TYPE	MAXIMUM DATA BLOCK LENGTH
2	6 (= 4 R/S ECC + 2 ECC EXT.)	SEC, DED	504 BYTES
2	10 (= 8 R/S ECC + 2 ECC EXT.)	DEC, TED	500 BYTES
3	9 (= 6 R/S ECC + 3 ECC EXT.)	SEC, DED	756 BYTES
3	15 (= 12 R/S ECC + 3 ECC EXT.)	DEC, TED	750 BYTES

SEC = single error correction.
DEC = double error correction.
TED = triple error detection.

It is also possible to program the redundancy bytes to exclude both an ECC extension and a CRC addition. In this case the total redundancy bytes, etc. take on the following form:

INTER-LEAVES	TOTAL REDUNDANCY BYTES	CORR TYPE	MAXIMUM DATA BLOCK LENGTH
2	4	SEC	506 BYTES
2	8	DEC	503 BYTES
3	6	SEC	759 BYTES
3	12	DEC	753 BYTES

SEC = single error correction.
DEC = double error correction.

LOCAL PROCESSOR INTERFACE BLOCK

This block has been optimized to interface with Local Processors that have a multiplexed 8 bit Address/Data bus. The internal local processor data bus and address bus are distributed to all blocks within the MSD95C01/02. Each block has its own address decoder as described in the OVERVIEW OF THE MSD95C01/02 REGISTERS section.

In addition, this block employs a complex interrupt structure used to generate appropriate interrupt based on multiple internal and external conditions. The interrupts can occur from three sources, namely the MICROSEQUENCER, the DMA block and via the four external sense inputs (SNS4-1) to the MSD95C01/02. Figure 8 illustrates the interrupt structure of the MSD92C01/02 with all the corresponding status and interrupt enable registers.

Overview of MSD95C01/02 Registers

Figure 7 shows the MSD95C01/02 address map. The 256 locations are internally decoded from the lower 8 bits of the address bus. Valid decode space exists from address 40H to FFH as shown. Address 00H to 2FH is not decoded by the MSD95C01/02 and can be used as register space for the External Device. The MSD95C00 SCSIC uses address 000H to 01FH for its internal register space, allowing the MSD95C00 and the MSD95C01/02 to share the same chip select decoded from the upper address bits of the local processor's address bus.

The MSD95C01/02 address space accessible to the local processor can be broken into four sections as follows:

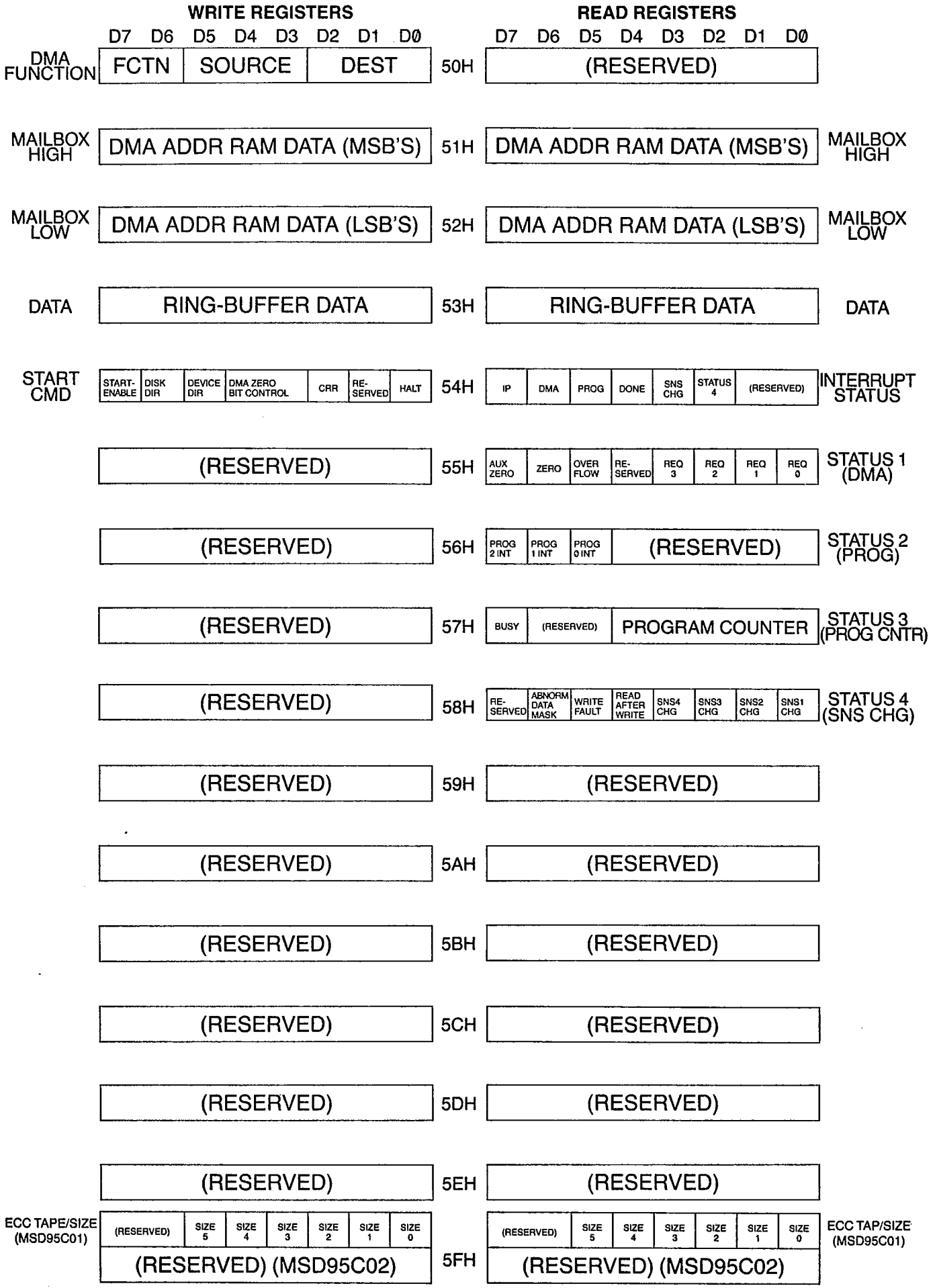
ADDRESS 40H-5FH

This address space contains the MSD95C01/02 working registers which include mode registers, setup registers, interrupt enable registers, status registers, and DMA parameter registers.

MSD95C01/02 REGISTERS

T-52-33-63

	WRITE REGISTERS D7 D6 D5 D4 D3 D2 D1 D0		READ REGISTERS D7 D6 D5 D4 D3 D2 D1 D0	
	(RESERVED)	40H	(RESERVED)	RESET
MODE 1	DMA CNTR RST | RE-SERVED | EX. DEVICE INCREMENT VALUE | CTL8-5 MODE | (RESERVED)	41H	(RESERVED)	
MODE 2	COMP/ INDIV | TAPE/ DISK | CLOCK SOURCE | CTL4 MODE | 16/8 COMP. | FORMAT	42H	(RESERVED)	
MODE 3	INV DATA | LEVEL/ PULSE | RL2SYN | (RESERVED) | CTL3-1 MODE	43H	(RESERVED)	
MODE 4	RE-SERVED | ECC PRESET (MSD95C01) INTER-LEAVE 2/3 (MSD95C02) | MAIN CRC | SUB CRC | CRC PRESET | RE-SERVED	44H	(RESERVED)	
MODE 5	(RESERVED)	45H	(RESERVED)	
SYNC 1	SYNC DATA	46H	(RESERVED)	
SYNC 2	SYNC DATA	47H	(RESERVED)	
SYNC 3	SYNC DATA	48H	(RESERVED)	
SYNC 4	SYNC DATA	49H	(RESERVED)	
TAPE HIGH BYTE	UNUSED | TAPE BYTE COUNTER	4AH	(RESERVED)	
TAPE LOW BYTE	TAPE BYTE COUNTER	4BH	(RESERVED)	
µC OUTPUT	CTL 8 | CTL 7 | CTL 6 | CTL 5 | CTL 4 | CTL 3 | CTL 2 | CTL 1	4CH	TST 4 | TST 3 | TST 2 | TST 1 | SNS 4 | SNS 3 | SNS 2 | SNS 1	µC INPUT
INT ENABLE 1	MAS | DMA | RE-SERVED | DONE | (RESERVED)	4DH	MAS | DMA | RE-SERVED | DONE | (RESERVED)	INT ENABLE 1
INT ENABLE 2	PROG 2 | PROG 1 | PROG 0 | RE-SERVED | RE-SERVED | RE-SERVED | RE-SERVED | RE-SERVED	4EH	PROG 2 | PROG 1 | PROG 0 | RE-SERVED | RE-SERVED | RE-SERVED | RE-SERVED | RE-SERVED	INT ENABLE 2
INT ENABLE 3	RE-SERVED | RE-SERVED | WRITE FAULT | READ AFTER WRITE | SNS 4 CHG | SNS 3 CHG | SNS 2 CHG | SNS 1 CHG	4FH	RE-SERVED | RE-SERVED | WRITE FAULT | READ AFTER WRITE | SNS 4 CHG | SNS 3 CHG | SNS 2 CHG | SNS 1 CHG	INT ENABLE 3


WRITE REGISTERS
READ REGISTERS
D7 D6 D5 D4 D3 D2 D1 D0
DMA FUNCTION
FCTN
SOURCE
DEST
50H
(RESERVED)
MAILBOX HIGH
DMA ADDR RAM DATA (MSB'S)
51H
MAILBOX LOW
DMA ADDR RAM DATA (LSB'S)
52H
DATA
RING-BUFFER DATA
53H
START CMD
START-ENABLE
DISK DIR
DEVICE DIR
DMA ZERO BIT CONTROL
CRR
RE-SERVED
HALT
54H
IP
DMA
PROG
DONE
SNS CHG
STATUS 4
(RESERVED)
INTERRUPT STATUS
55H
AUX ZERO
ZERO
OVER FLOW
RE-SERVED
REQ 3
REQ 2
REQ 1
REQ 0
STATUS 1 (DMA)
56H
PROG 2 INT
PROG 1 INT
PROG 0 INT
STATUS 2 (PROG)
57H
BUSY
PROGRAM COUNTER
STATUS 3 (PROG CNTR)
58H
ABNORM DATA MASK
WRITE FAULT
READ AFTER WRITE
SNS4 CHG
SNS3 CHG
SNS2 CHG
SNS1 CHG
STATUS 4 (SNS CHG)
59H
5AH
5BH
5CH
5DH
5EH
ECC TAPE/SIZE (MSD95C01)
SIZE 5
SIZE 4
SIZE 3
SIZE 2
SIZE 1
SIZE 0
(RESERVED) (MSD95C02)
5FH
ECC TAP/SIZE (MSD95C01)

ADDRESS 60H-6FH

This address space contains an on-chip 16 byte register file referred to as the DESIRED REGISTER FILE. It is shared by the local processor and the MICROSEQUENCER and is typically loaded by the local processor and internally compared with data from the disk. Local processor access to this address space should only occur when the MICROSEQUENCER is not running.

ADDRESS 70H-FFH

This address space contains an on chip, 16 byte, register file referred to as the CURRENT REGISTER FILE. It is shared by the local processor and the MICROSEQUENCER and is typically loaded with data from the disk for examination later by the local processor. Local processor access to this address space is only allowed during certain MICROSEQUENCER initiated interrupts to the local processor.

ADDRESS 80H-8FH

This address space contains the 128 bytes of loadable microcode used by the MICROSEQUENCER. Internally, these bytes are arranged as 32 locations by 32 bits wide. Local processor address 80H corresponds to the least significant byte (D7-0) of word zero of the microcode RAM. This address space should not be accessed by the local processor while a microprogram is running.

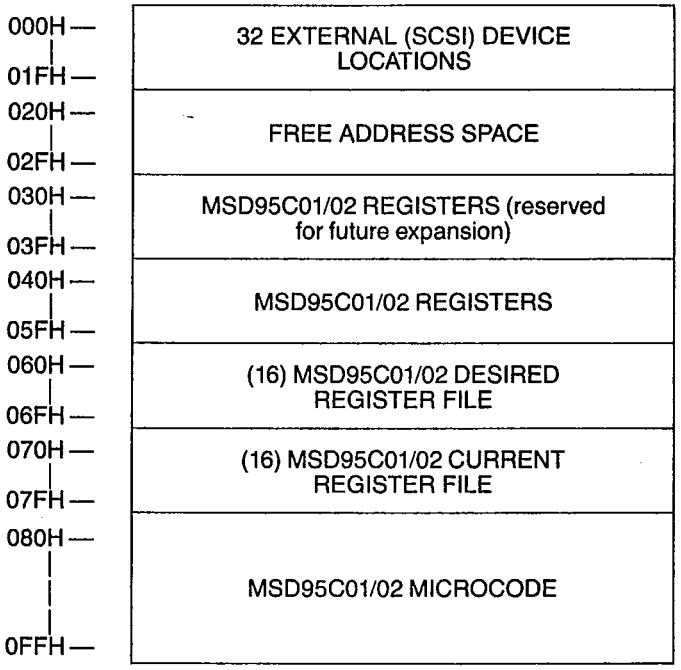


FIGURE 7: MSD95C01/02 ADDRESS MAP

Register Descriptions

All reserved bits are read as zero and all unused bits should be written as zero for compatibility with future options.

RESET (READ)—40H

Reading this register address will reset the MSD95C01/02. This is referred to as a "soft" reset. Soft and hard resets are indistinguishable. Reset will cause the following I/O pins to be forced to the states indicated:

DB7-0	Input
OE	Inactive High
WE	Inactive High
DACK	Inactive High
RGATE	Inactive Low
WGATE	Inactive Low
AD7-0	Input
INT	Floats
CTL1 (EARLY)	Inactive Low
CTL2 (LATE)	Inactive Low
CTL3 (WRFCLK)	Input
CTL4 (AMEN)	Inactive Low

T-52-33-63

Reset will be terminated by a write to any MSD95C01/02 register.

Reserved Register (WRITE)—40H

This register space is not presently implemented.

MODE 1 Register (WRITE)—41H

The bits in this register are cleared to "0" by a hard or soft reset. This Mode Register is used to control the DMA section.

BIT D7: DMA TRANSFER REQUEST COUNTER RESET

Writing a "1" to this bit causes the DMA transfer request counter to be reset. This bit does not have to be cleared between writing successive "1's."

BIT D6: RESERVED

BIT D5-D4: EXTERNAL DEVICE DMA ADDRESS INCREMENT VALUE

These bits determine the value that the External Device's DMA address will be incremented for every External Device DMA cycle:

D5	D4	INCREMENT VALUE
0	0	1
0	1	2
1	0	3
1	1	4

BIT D3: CONTROL OUTPUTS MODE SELECT

These bits determine the function of the output pins CTL8-5:

D3 = 0: CTL8 is DGATE.
CTL7 is WRITE.
CTL6 is ST0.
CTL5 is ST1.

D3 = 1: CTL8-5 are local processor programmed outputs.
BITS D2-D0: RESERVED

D3 = 1: CTL8-5 are local processor programmed outputs.
BITS D2-D0: RESERVED

MODE 2 Register (WRITE)—42H

This Mode Register is used to control the Disk Interface and Encoder/Decoder section.

BIT D7: COMPOSITE/INDIVIDUAL SYNDROME

D7 = 0: The ECC uses individual syndrome.
D7 = 1: The ECC uses composite syndrome.

BIT D6: TAPE/DISK MODE

D6 = 0: Select disk mode.
D6 = 1: Select tape mode, perform Read after Write using CRC.

BIT D5: REFERENCE CLOCK SOURCE

This bit is used to select the clock source for the bit rate Read/Write circuitry used when the Read Gate is inactive. (This bit must be a "1" when using Read after Write for tape mode.)

T-52-33-63

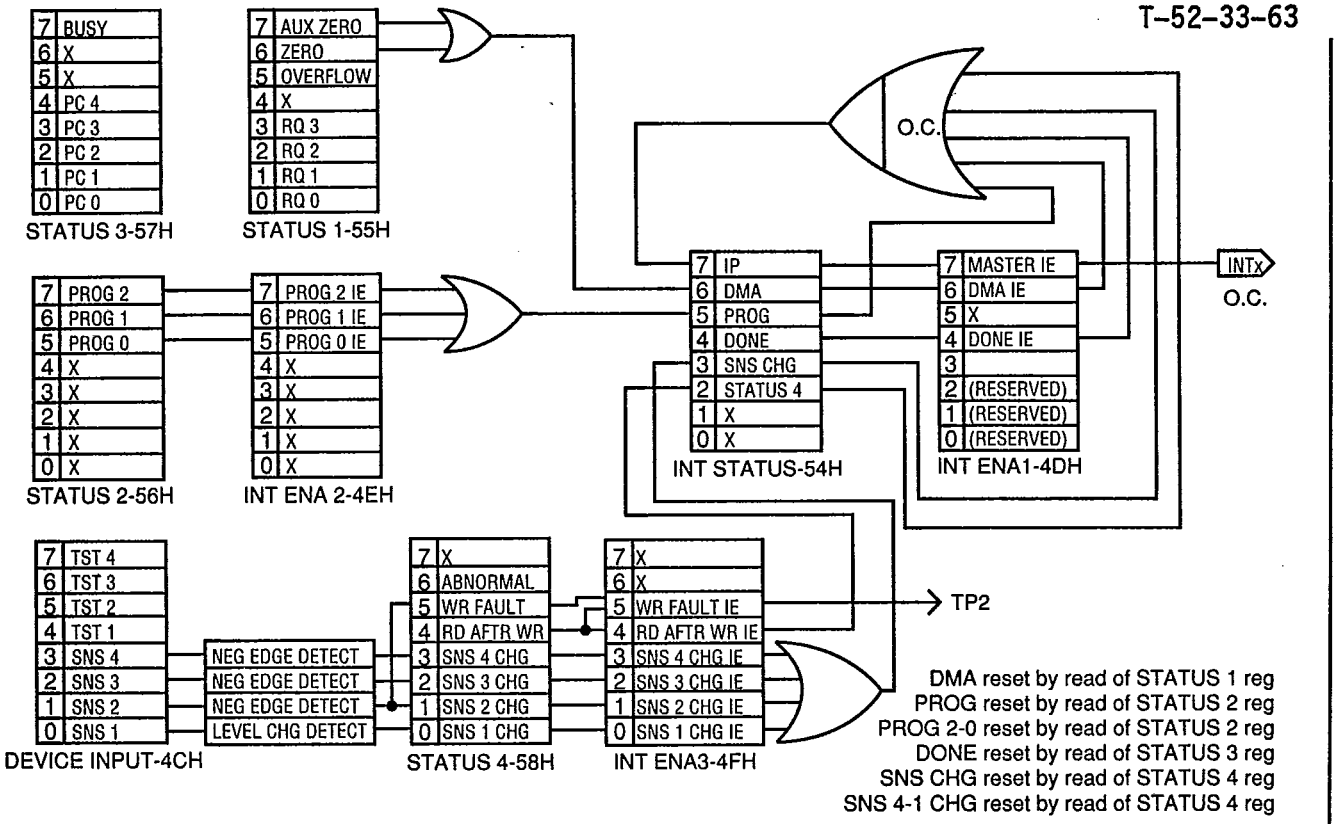


FIGURE 8

D5 = 0: RCLK (COMBINED READ/REFERENCE CLOCK)
D5 = 1: WRITE REF CLOCK. May be selected for Programmable ECC (MSD95C01), Reed Solomon ECC (MSD95C02), or 16-bit internal CRC, but not for 32-bit internal CRC.

BIT D4: CONTROL OUTPUTS MODE SELECT

These bits determine the function of the CTL4 pin:

D4 = 0: CTL4 is MICROSEQUENCER output (typically used for ESDI AM enable)
D4 = 1: CTL4 is Local Processor output pin

BIT D3: 16/8 BIT COMPARISON

This bit will choose the 16 or 8 bit compare mechanism in the ENCODER/DECODER block. The 8 bit compare is used when accepting data in NRZ format. The 16 bit compare is used when accepting data in an encoded mode typically to establish bit synchronization. Refer to the SYNC Register description for a table of how the compares interact with this bit.
D3 = 0: Choose 8 bit compare.
D3 = 1: Choose 16 bit compare.

BITS D2-D0: FORMAT

These bits are used to control the Encoder and Decoder.

D2	D1	D0	TYPE OF ENCODING
0	0	0	FM
0	0	1	MFM
0	1	0	UNUSED
0	1	1	UNUSED
1	0	0	RLL 2, 7 TYPE 1
1	0	1	RLL 2, 7 TYPE 2
1	1	0	GCR
1	1	1	NRZ

TABLE 2: ENCODER/DECODER SELECTION

Tables 3 and 4 show the correspondence mapping for RLL and GCR encoding respectively.

INPUT BIT STREAM	RLL 2, 7 TYPE 1	RLL 2, 7 TYPE 2
10	0100	0100
010	000100	100100
0010	00100100	00100100
11	1000	1000
011	001000	001000
0011	00001000	00001000
000	100100	000100

TABLE 3: RLL ENCODING MAPPING

INPUT BIT STREAM	GCR ENCODING
0000	11001
0001	11011
0010	10010
0011	10011
0100	11101
0101	10101
0110	10110
0111	10111
1000	11010
1001	01001
1010	01010
1011	01011
1100	11110
1101	01101
1110	01110
1111	01111

TABLE 4: GCR ENCODING MAPPING

MODE 3 Register (WRITE)—43H

This Mode Register is used to control the DISK INTERFACE and ENCODER/DECODER blocks.

BIT D7: INVERT DATA

Setting this bit to "1" will cause the disk data to be inverted before encoding and after decoding. If this bit is "0", the data will not be inverted.

BIT D6: LEVEL/PULSE

Setting this bit to "1" will cause the disk data to be decoded as hard disk data (level transitions). If this bit is "0", the data will be decoded as floppy disk (pulse) data.

BIT D5: RL2SYN

This bit should be set high for RLL2 with preamble 100, otherwise it should be set low.

BITS D4-D2: RESERVED.

BITS D1-D0: CONTROL OUTPUTS MODE SELECT

These bits determine the function of the CTL3-1 pins:

D1 = 0: CTL3 is WRFCLK input
D1 = 1: CTL3 is Local Processor output pin
(Note: bit D1 is set to a "0" upon a hard or soft reset)
D0 = 0: CTL2 is LATE output
CTL1 is EARLY output
D0 = 1: CTL2 and 1 are Local Processor programmable outputs.

MODE 4 Register (WRITE)—44H

This Mode Register is used to control the ECC Interface section. Note that bits D6-D3 perform different functions in the MSD95C02 than they do in the MSD95C01.

BIT D7: RESERVED in both the MSD95C01 and the MSD95C02

D6: ECC/PRESET or INTERLEAVE

For the MSD95C01, this bit initializes the ECC Register. For the MSD95C02, this bit selects the interleave.

D6	MSD95C01	MSD95C02
0	Initialize Programmable ECC Register to all O's before computing.	Select interleave 3.
1	Initialize Programmable ECC Register to all 1's before computing.	Select interleave 2.

TABLE 5: ECC REGISTER INITIALIZATION/ INTERLEAVE SELECTION

BITS D5-D3: MAIN CRC OPTION

These bits select the error correcting scheme used when MAIN is selected by the microcode. See Table 6.

BIT D2:SUB CRC OPTION

This bit selects the CRC option when SUB is selected by the microcode in both the MSD95C01 and the MSD95C02.

D2 = 1: 32-bit ECC, IBM AT

D2 = 0: 16-bit CRC

BIT D1: CRC/PRESET

This bit initializes the 16-bit CRC/32-bit ECC Register in both the MSD95C01 and the MSD95C02.

D1 = 1: Initialize 16-bit CRC/32-bit ECC Register to "1" before computing.
D1 = 0: Initialize 16-bit CRC/32-bit ECC Register to "0" before computing.

BIT D0: RESERVED in both the MSD95C01 and the MSD95C02.

D5	D4	D3	MSD95C01	MSD95C02
0	0	0	16-bit CRC.	16-bit CRC.
0	0	1	Programmable ECC, reset feedback tap pointer, enable tap programming, exclude sync mark.	Reed Solomon, degree 4.
0	1	0	Programmable ECC, enable ECC length programming, exclude sync mark.	Reed Solomon, degree 4 and extension.
0	1	1	Invalid.	Reed Solomon, degree 4 and 24-bit CRC.
1	0	0	32-bit ECC, IBM AT	32-bit ECC, IBM AT.
1	0	1	Programmable ECC, reset feedback tap pointer, enable tap programming, include sync mark.	Reed Solomon, degree 2.
1	1	0	Programmble ECC, enable ECC length programming, include sync mark.	Reed Solomon, degree 2 and extension.
1	1	1	Invalid.	Reed Solomon, degree 2 and 24-bit CRC.

TABLE 6: ECC CODE OPTIONS

RESERVED REGISTER SPACE—45H

This register space is reserved for future expansion.

SYNC REGISTER 1-4 (WRITE)—ADDR 46H-49H.

The following four SYNC Registers are used to perform either 8 or 16 bit high speed compares on the incoming encoded data from the disk. The compares are executed via three signals in the microcode EXT field. The three signals, SPRE, SMC and SDM are typically used to look for PLO SYNC (preamble) data, missing clock pattern data, and data mark data respectively. The MICROSEQUENCER can check for a valid high speed compare via the signal SYNC in the test field of the microcode word. Tables 7 and 8 illustrate how the compares are used in conjunction with the four SYNC Registers as a function of the encoding scheme used and bit D3 (8/16 bit compare) of MODE 2 Register.

The following two registers represent a 12 bit TAPE BYTE counter. This counter is needed during tape Read/Write operating to inform the readback circuitry that the data block plus the two byte CRC field has been read. Only the CRC-16 can be used with the tape feature. It is loaded with the value equal to the number of bytes in the data block plus 2 (for the CRC-16 field).

TAPE BYTE COUNTER HIGH (WRITE)—4AH

This register holds the upper four bits of the twelve bit Tape Byte Counter. This counter is used to count the number of bytes used for CRC 2 during the Read after Write.

BITS D7-D4: RESERVED

BITS D3-D0: Upper four bits of Tape Byte Counter.

TAPE BYTE COUNTER LOW (WRITE)—4BH

This register holds the lower eight bits of the twelve bit Tape Byte Counter. This counter is used to count the number of bytes used for CRC 2 during the Read after Write.

REGISTERS (READ)—59H, 5AH, 5BH

These registers are reserved for future expansion.

T-52-33-63

SYNC 1 Register (WRITE)—46H This register holds SYNC 1.

SYNC 2 Register (WRITE)—47H This register holds SYNC 2.

SYNC 3 Register (WRITE)—48H This register holds SYNC 3.

SYNC 4 Register (WRITE)—49H This register holds SYNC 4.

ENCODING SCHEME	SYNC 1 (46H)	SYNC 2 (47H)	SYNC 3 (48H)	SYNC 4 (49H)
FM	ADDRESS MARK DATA	DATA MARK DATA	PLO SYNC DATA	DATA MARK CLOCK
MFM	MISSING CLOCK DATA	PLO SYNC DATA	PLO SYNC CLOCK	MISSING CLOCK CLOCK
RLL (Note 1)	MISSING CLOCK DATA	PLO SYNC DATA	PLO SYNC CLOCK	MISSING CLOCK CLOCK
RLL (Note 2)	—	—	PREAMBLE DATA	SYNC MARK
NRZ (Note 3)	—	—	DATA MARK	ADDRESS MARK
GCR (Note 3)	BLOCK MARK	PREAMBLE DATA	—	—

TABLE 7: SYNC REGISTER DEFINITION AS A FUNCTION OF ENCODING SCHEME

(Note 1) RLL SYNC Register setup for 16 bit compare (bit D3 of MODE 2 Register = 1)
(Note 2) RLL SYNC Register setup for 8 bit compare (bit D3 of MODE 2 Register = 0)
(Note 3) NRZ and GCR encoding should always use 8 bit compares (bit D3 of MODE 2 Register = 0)

MICROCODE SIGNAL	8/16 COMPARE	COMPARE BETWEEN DISK DATA AND SYNC REGISTER
SPRE	8	SYNC 2 (47H) FOR GCR; SYNC 3 (48H) FOR ALL OTHERS
SPRE	16	SYNC 2 (47H) FOR DATA TRACK AND SYNC 3 (48H) FOR CLOCK TRACK
SAM	8	SYNC 1 (46H) FOR GCR; SYNC 4 (49H) FOR ALL OTHERS
SAM	16	SYNC 1 (46H) FOR DATA TRACK AND SYNC 4 (49H) FOR CLOCK TRACK
SDM	8	SYNC 2 (47H) FOR GCR; SYNC 4 (49H) FOR ALL OTHERS
SDM	16	SYNC 2 (47H) FOR DATA TRACK AND SYNC 4 (49H) FOR CLOCK TRACK

TABLE 8: SYNC REGISTER COMPARE MATRIX AS A FUNCTION OF MICROCODE SIGNALS

DEVICE INPUTS Register (READ)—4CH

This register reflects the status of the external TEST and SENSE inputs. The MSD95C01/02 can be programmed to generate an interrupt from some of these input pins as follows:

BIT D7: TEST 4
This bit reflects the state of the TST4 input.

BIT D6: TEST 3
This bit reflects the state of the TST3 input.

BIT D5: TEST 2
This bit reflects the state of the TST2 input.

BIT D4: TEST 1
This bit reflects the state of the TST1 input.

BIT D3: SENSE 4
This bit reflects the state of the SNS4 input. The MSD95C01/02 can be programmed to generate an interrupt whenever this input transitions from high to low (negative edge triggered interupt).

BIT D2: SENSE 3
This bit reflects the state of the SNS3 input. The MSD95C01/02 can be programmed to generate an interrupt whenever this input transitions from high to low (negative edge triggered interrupt).

BIT D1: SENSE 2
This bit reflects the state of the SNS2 input. The MSD95C01/02 can be programmed to generate an interrupt whenever this input transitions from high to low (negative edge triggered interrupt).

BIT D0: SENSE 1
This bit reflects the state of the SNS1 input. The MSD95C01/02 can be programmed to generate an interrupt whenever this input transitions (level change interrupt).

LOCAL PROCESSOR OUTPUT Register (WRITE)—4CH

This register holds the data that is directly output on the CTL8-1 pins when they are configured as Local Processor outputs by bits in MODE 1, MODE 2 and MODE 3 Registers. Bit 7 is output on CTL8 and bit 0 is output on CTL1. These bits are cleared to zero by a hard or soft reset.

INTERRUPT ENABLE 1 Register (READ/WRITE)—4DH

The bits in this register are cleared to zero by a hard or soft reset.

BIT D7: MASTER INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to drive its $\overline{INT}$ output pin active when an enabled condition causes the Interrupt Pending status bit to go active high.

BIT D6: DMA INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Interrupt Pending status bit when the DMA bit in the INTERRUPT STATUS Register goes active high.

BIT D5: (RESERVED)

BIT D4: DONE INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Interrupt Pending status bit when the DONE bit in the INTERRUPT STATUS Register goes active high.

BIT D3: (RESERVED)

BIT D2-D0: (reserved for future expansion)

INTERRUPT ENABLE 2 Register (READ/WRITE)—4EH

The bits in this register are cleared to zero by a hard or soft reset.

BIT D7: PROGRAM 2 INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Interrupt Pending and Program status bits in the INTERRUPT STATUS Register when the PROG 2 bit in the STATUS 2 Register goes active high.

BIT D6: PROGRAM 1 INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Interrupt Pending and Program status bits in the INTERRUPT STATUS Register when the PROG 1 bit in the STATUS 2 Register goes active high.

BIT D5: PROGRAM 0 INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Interrupt Pending and Program status bits in the INTERRUPT STATUS Register when the PROG 0 bit in the STATUS 2 Register goes active high.

BITS D4-D0: (RESERVED)

INTERRUPT ENABLE 3 Register (READ/WRITE)—4FH

The bits in this register are cleared to zero by a hard or soft reset.

BITS D7-D6: MUST BE SET TO ZERO (RESERVED)

BIT D5: WRITE FAULT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Write Fault bit in the STATUS 4 Register and cause the test point "NTP2" to go active when the Sense 2 Change bit in the STATUS 4 Register goes active high.

BIT D4: READ AFTER WRITE INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Interrupt Pending and STATUS 4 Interrupt bits in the INTERRUPT STATUS Register when the Read After Write Fault bit in the STATUS 4 Register goes active high.

BIT D3: SENSE 4 CHANGE INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Interrupt Pending and Sense Change bits in the INTERRUPT STATUS Register when the Sense 4 Change bit in the STATUS 4 Register goes active high.

BIT D2: SENSE 3 CHANGE INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Interrupt Pending and Sense Change bits in the INTERRUPT STATUS Register when the SENSE 3 Change bit in the STATUS 4 Register goes active high.

BIT D1: SENSE 2 CHANGE INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Interrupt Pending and Sense Change bits in the INTERRUPT STATUS Register when the SENSE 2 Change bit in the STATUS 4 Register goes active high.

BIT D0: SENSE 1 CHANGE INTERRUPT ENABLE
Setting this bit to "1" will cause the MSD95C01/02 to set the Interrupt Pending and Sense Change bits in the INTERRUPT STATUS Register when the SENSE 1 Change bit in the STATUS 4 Register goes active high.

DMA FUNCTION Register (WRITE)—50H

BITS D7-D6. These bits specify the operation to be performed on the internal register file by the Local Processor:

D7	D6	FUNCTION
0	0	SOURCE + OFFSET COUNTER → DESTINATION (Note 1)
0	1	SOURCE + N → DESTINATION; DECREMENT OFFSET COUNTER
1	0	SOURCE → DESTINATION
1	1	SOURCE + N → DESTINATION

N = 1 for disk and local processor data transfers
N specified for MODE 1 Register for External channel (SCSI) data transfers

TABLE 9: DMA FUNCTION REGISTER OPERATION

Notes:
1. For this case, the only destinations that can be updated are the OFFSET COUNTER and the MAILBOX. All other destinations will cause the OVERFLOW flag to be updated as a function of the result out of the ALU, but will not update the destination.

2. When the controller changes the contents of the OFFSET and AUXILIARY OFFSET COUNTERS, it must set or clear the zero bit via the START COMMAND Register bits 4 and 3.

3. The OFFSET COUNTER cannot be used as a source. If it is used as a source, it will take on the value of zero. During error correction, it becomes necessary to use the OFFSET COUNTER as a source. See appendix 3 for a description of ERROR CORRECTION ON THE FLY for more detail.

BITS D5-D3. Source register file address.
BITS D2-D0. Destination register file address.

These bits select the register file address location(s) accessed by the operation specified by bits D7 and D6.

MAILBOX HIGH Register (READ/WRITE)—51H

BITS D7-D0: HIGH ORDER DMA ADDRESS BITS; A15-8

This register functions as a mailbox for transfer of the high order address bits to and from the DMA register file.

T-52-33-63

D5 D2	D4 D1	D3 D0	REGISTER SELECTED
0	0	0	MAILBOX HIGH, LOW
0	0	1	OFFSET COUNTER
0	1	0	CONSTANT 1
0	1	1	CONSTANT 2
1	0	0	LOCAL PROCESSOR ADDRESS
1	0	1	EXTERNAL DEVICE ADDRESS
1	1	0	DISK REGISTER
1	1	1	DISK ADDRESS

TABLE 10: DMA FUNCTION REGISTER SOURCE/ DESTINATION

MAILBOX LOW Register (READ/WRITE)—52H

BITS D7-D0: LOW ORDER DMA ADDRESS BITS; A7-0

This register functions as a mailbox for transfer of the low order address bits to and from the DMA register file.

DATA Register (READ/WRITE)—53H

This register functions as a mailbox for data transfers between the Local Processor and the Ring Buffer. Local Processor read and writes to the Ring Buffer are funnelled through this register which resides in the DMA block. The Local Processor should poll STATUS 1 Register, bit 1 (REQUEST 1), to determine the status of local processor initiated DMA cycles.

START COMMAND Register (WRITE)—54H

Writing to this register initiates microprogram execution.

BIT D7: START ENABLE

Setting this bit to a logic "1" will set the BUSY bit in STATUS 3 Register and enable the start of microprogram execution.

BIT D6: DISK DMA DIRECTION
This bit selects the direction of DMA data transfer for the disk data channel:

D6 = 1 Disk to Ring Buffer (Disk READ)
D6 = 0 Ring Buffer to Disk (Disk WRITE)

BIT D5: EXTERNAL DEVICE DMA DIRECTION
This bit selects the direction of DMA data transfer for the EXTERNAL data channel:

D5 = 1 External Device to Ring Buffer.
D5 = 0 Ring Buffer to External Device.

BIT D4, D3: DMA ZERO BIT CONTROL
These bits can set or reset the zero output of the DMA OFFSET and AUXILIARY OFFSET COUNTERs:

D3	D4	RESULT
0	0	No change to zero indicator
0	1	Reset zero indicator
1	0	Set zero indicator
1	1	Undefined

TABLE 11: DMA ZERO BIT CONTROL

BIT D2: CRR (Current Register Read)
Writing a logic "1" to this register indicates that the microprocessor has finished reading the information in the CURRENT REGISTER FILE. This must be done after an interrupt has been caused by PROG0 or PROG1 to allow MICROSEQUENCER access to the CURRENT REGISTER FILE. Writing a logic zero will have no effect.

BIT D1: (RESERVED)

BIT D0: HALT
Writing a logic "1" to this register will cause "NTP2" to go active. The MICROSEQUENCER can test "NTP2" at certain times to determine if it should terminate execution of the present command. This bit is a way for the local processor to halt the microprogram at an appropriate time (for example; at the end of a sector read) by having the microprogram interrogate NTP2 at that time.

INTERRUPT STATUS Register (READ)—54H

This register is the master interrupt status register. It should be read after an interrupt is generated to determine which logical block generated the interrupt condition.

BIT D7: INTERRUPT PENDING
This bit is set to "1" when one of the enabled interrupt conditions occur. It is cleared to "0" when the interrupt causing condition(s) are cleared.

BIT D6: DMA INTERRUPT
This bit is set to "1" when the ZERO or AUXILIARY ZERO bits in the STATUS 1 (DMA) Register go active high. It is cleared to "0" by reading the STATUS 1 (DMA) Register, a hard or a soft reset.

BIT D5: PROGRAM INTERRUPT
This bit is set to "1" when one of the enabled PROG2-0 bits in the STATUS 2 Register goes active high. It is cleared to "0" by reading the STATUS 2 Register, a hard or a soft reset.

BIT D4: DONE INTERRUPT
This bit is set to "1" when the MSD95C01/02 completes a command. This interrupt is generated by the microcode and indicates that the internal MICROSEQUENCER has stopped. This bit is reset to "0" by reading STATUS 3 Register, or by a hard or soft reset.

BIT D3: SENSE CHANGE INTERRUPT
This bit is set to "1" when one of the enabled Sense 4-1 Change bits in the STATUS 4 Register goes active high. This bit is reset to "0" by reading the STATUS 4 Register, a hard or a soft reset.

BIT D2: STATUS 4 Register INTERRUPT
This bit is set to "1" when the Read After Write interrupt in STATUS 4 Register goes active high. This bit is reset to "0" by reading the STATUS 4 Register, a hard or a soft reset.

BITS D1-D0: (RESERVED)

STATUS 1 Register (READ)—55H

This register contains status about the DMA block. Bits D3-D0 are cleared to zero following a hard or soft reset. Bits D7-D4 are unaffected by a hard or soft reset.

BIT D7: AUXILIARY ZERO
This bit reflects the state of the zero flag of the AUXILIARY OFFSET COUNTER.

BIT D6: ZERO
This bit reflects the state of the zero flag of the OFFSET COUNTER.

BIT D5: OVERFLOW
This bit reflects the state of the overflow flag generated from the ALU in the DMA block.

BIT D4: (RESERVED)

BIT D3: REQUEST 3
This bit reflects the state of the MICROSEQUENCER housekeeping DMA cycle request latch. A logic "1" indicates that the DMA cycle request has either not been acknowledged or is in progress and a logic "0" indicates that the DMA cycle has been completed.

BIT D2: REQUEST 2
This bit reflects the state of the MICROSEQUENCER data transfer DMA cycle request latch. A logic "1" indicates that the DMA cycle request has either not been acknowledged or is in progress and a logic "0" indicates that the DMA cycle has been completed.

BIT D1: REQUEST 1
This bit reflects the state of the Local Processor channel DMA cycle request latch. A logic "1" indicates that the DMA cycle request has either not been acknowledged or is in progress and a logic "0" indicates that the DMA cycle has been completed.

BIT D0: REQUEST 0
This bit reflects the state of the External Device channel DMA cycle request latch. A logic "1" indicates that the DMA cycle request has either not been acknowledged or is in progress and a logic "0" indicates that the DMA cycle has been completed.

STATUS 2 Register (READ)—56H

This register contains status about the MICROSEQUENCER block. The Program Interrupt bits are set under MICROSEQUENCER control. The PROG2-0 bits are cleared to zero by reading the STATUS 2 Register, a hard or a soft reset.

BITS D7-D5: PROGRAMMABLE INTERRUPTS 2-0.
These bits are set to "1" under MICROSEQUENCER control. These bits are reset to "0" by reading the Status 2 Register, a hard or a soft reset.

BIT D7: PROGRAMMABLE INTERRUPT 2
This is a programmable interrupt that is set under a MICROSEQUENCER control. This is a general purpose interrupt that can be generated upon recognition of any MICROSEQUENCER detectable condition. The generation of this interrupt will affect no additional hardware.

BIT D6: PROGRAMMABLE INTERRUPT 1
This is a programmable interrupt that is set under MICROSEQUENCER control. This interrupt will allow the local processor to access the CURRENT REGISTER FILE. The MICROSEQUENCER will not be able to update the register file until the local processor has indicated that it has finished reading the register file by writing a "1" to bit D2 (CRR) of the START COMMAND Register. This interrupt may be used for ID field interrupt to permit the local processor to read the current ID information that is stored

BIT D5: PROGRAMMABLE INTERRUPT 0
This is a programmable interrupt that is set under MICROSEQUENCER control. This interrupt will allow the local processor to access the CURRENT REGISTER FILE. The MICROSEQUENCER will not be able to update the CURRENT REGISTER FILE until the local processor has indicated that it has finished reading the CURRENT REGISTER FILE by writing a "1" to bit D2 (CRR) of the START COMMAND Register. The generation of this interrupt will delink the OFFSET and AUXILIARY OFFSET COUNTERS in the DMA block. This interrupt is used to indicate an ECC error. Service would be required by having the Local Processor read the error syndrome residing in the CURRENT REGISTER FILE, correct the error and properly relink the AUXILIARY OFFSET and OFFSET COUNTERs in the DMA block back together. See Appendix 4 (ECC ON THE FLY) for further details.

BITS D4-D0: (RESERVED)

STATUS 3 Register (READ)—57H

This register contains status about the MICROSEQUENCER block. The values in bits D4-0 are valid when the Done Interrupt bit is active high.

BIT D7: BUSY
This bit indicates the state of the MICROSEQUENCER.
0 = MICROSEQUENCER is finished or reset.
1 = MICROSEQUENCER is executing a program.

BITS D6-D5: (RESERVED)

BITS D4-D0: PROGRAM COUNTER DATA
These bits will hold the address of the instruction executed prior to executing the instruction that generated the Done Interrupt via the signal DNINT in the OUT field of the MICROSEQUENCER RAM. They are valid after the Done status bit is set indicating that the MICROSEQUENCER has stopped running.

STATUS 4 Register (READ)—58H

This register contains status information.

BIT D7: RESERVED

BIT D6: ABNORMAL DATA MARK
This bit is set to "1" when the MICROSEQUENCER tests the data mark and it does not match the data mark in the DESIRED REGISTER. This bit being set will cause "NTP2" to become active. This bit is reset by the start of execution of a command.

BIT D5: WRITE FAULT
This bit is set to "1" when the SNS2 input undergoes a high-to-low level transition if the WRITE FAULT enable bit of the INTERRUPT ENABLE 3 Register is an active "1". This bit being set will cause "NTP2" to become active. This bit is reset to "0" by reading the STATUS 4 Register, a hard or a soft reset.

BIT D4: READ AFTER WRITE ERROR
This bit is set to "1" when a CRC error occurs during a Read After Write while in the tape mode. This bit is reset to "0" by reading the STATUS 4 Register, a hard or a soft reset. This bit may be set while not in tape mode, but should only be checked in tape mode.

BIT D3: SENSE 4 CHANGE
This bit is set to "1" when the SNS4 input undergoes a high-to-low level transition. This bit is reset to "0" by reading the STATUS 4 Register, a hard or a soft reset.

BIT D2: SENSE 3 CHANGE
This bit is set to "1" when the SNS3 input undergoes a high-to-low level transition. This bit is reset to "0" by reading the STATUS 4 Register, a hard or a soft reset.

BIT D1: SENSE 2 CHANGE
This bit is set to "1" when the SNS2 input undergoes a high-to-low level transition. This bit is reset to "0" by reading the STATUS 4 Register, a hard or a soft reset.

BIT D0: SENSE 1 CHANGE
This bit is set to "1" when the SNS1 input undergoes a high-to-low or low-to-high level transition. This bit is reset to "0" by reading the STATUS 4 Register, a hard or a soft reset.

REGISTERS (READ)—59H, 5AH, 5BH

These registers are reserved for future expansion.

FEEDBACK TAP/SIZE Register (READ/WRITE)—5FH

This register is used to program the feedback locations or size to implement the desired ECC polynomial. This register exists only in the MSD95C01.

FEEDBACK TAP

T-52-33-63

When the Mode 4 register bits D5-D3 are set to enable tap programmming, recursive writes to register 5FH will program a feedback tap register starting with the most significant byte and ending on the least significant byte. Only the number of bytes needed to implement the desired polynomial need to be programmed. If more than eight successive writes to the feedback tap register (5FH) occur, the last value will reside in the least significant tap register. Only an intervening Mode 4 (44H) write to reset the feedback tap pointer will enable the recursive write procedure to the most significant tap register. When the polynomial length is not a multiple of eight, only some bits in the most significant feed tap register are used.

SIZE

When the Mode 4 register bits D5-D3 are set to enable ECC length programming, writing register 5FH will program the size and set the highest degree term of the polynomial. The value must be the length of the polynomial minus one.

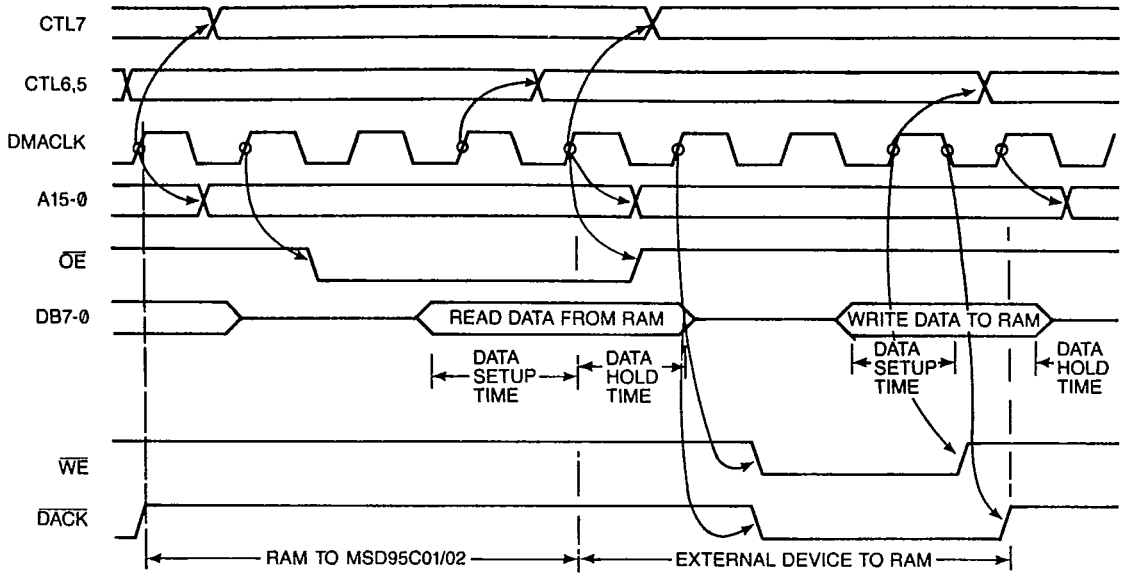


FIGURE 9: TYPICAL BUFFER TIMING (DISK WRITE)

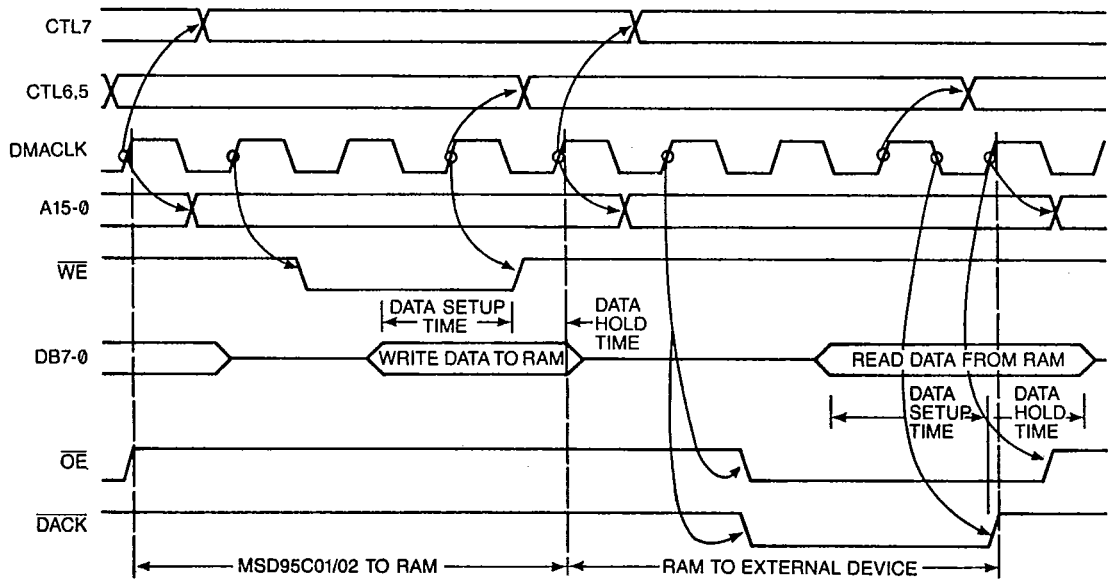


FIGURE 10: TYPICAL BUFFER TIMING (DISK READ)

OPERATIONAL DESCRIPTION

T-52-33-63

MAXIMUM GUARANTEED RATINGS

Operating Temperature Range ..... 0° to 70°C
Storage Temperature Range ..... −55°C to +150°C
Lead Temperature (soldering, 10 sec) ..... +300°C
Positive Voltage on any Pin, with respect to Ground ..... V_{CC} +0.3V
Negative Voltage on any Pin, with respect to Ground ..... −0.3V
Maximum Voltage on V_{CC} pin 7.0V

Stresses above those listed may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied.

NOTE: When powering this device from laboratory or system power supplies, it is important that the "Maximum Guaranteed Ratings" not be exceeded, or device failure can result. Some power supplies exhibit voltage spikes or "glitches" on their outputs when AC power is switched off. In addition, voltage transients on the AC power line may appear on the DC output. If this possibility exists it is suggested that a clamp circuit be used.

DC ELECTRICAL CHARACTERISTICS T_A = 0°C to 70°C, V_{CC} = 5.0V, ±5%)

PRELIMINARY
Notice: This is not a final specification.
Some parametric limits are subject to change.

PARAMETER	MIN	TYP	MAX	UNITS	COMMENTS
SUPPLY CURRENT					
I_{CC}		80		mA	V_{CC} = 5.25V
$I_{STANDBY}$		0.2		mA	V_{CC} = 5.25V
OUTPUT VOLTAGE					
High, V_{OH}	2.4			V	I_{OH} = −40µA
Low, V_{OL}			0.4	V	I_{OL} = 1.6mA
INPUT VOLTAGE					
High, V_{IH1}	2.0			V	Except DMACLK, RCLK, CTL3
High, V_{IH2}	3.5			V	DMACLK, RCLK, CTL3
Low, V_{IH1}			0.8	V	Except DMACLK, RCLK, CTL3
Low, V_{IH2}			1.5	V	DMACLK, RCLK, CTL3
INPUT LEAKAGE					
High, I_{IH}			10	µA	V_{IH} = 2.0V, Except DMACLK, RCLK, CTL3
Low, I_{IL}			10	µA	V_{IH} = 3.5V, DMACLK, RCLK, CTL3
					V_{IL} = 0.8V, Except DMACLK, RCLK, CTL3
					V_{IL} = 1.5V, DMACLK, RCLK, CTL3

AC ELECTRICAL CHARACTERISTICS

All minimum and maximum times are given assuming a 20MHz clock. If the time is clock dependent, then the equation is given under the comments column. This is to allow calculation of the time delays using slower clock rates.

AC ELECTRICAL SPECIFICATIONS T_A = 0°C to 70°C, V_{CC} = 5.0V, ±5%)

DATA PATH TIMING

SYMBOL	MIN	MAX	UNITS	COMMENTS
T_1	25		ns	ADDR TO $\overline{OE}$ (DMACLK) −25 nsec
T_2	0		ns	READ DATA HOLD FROM $\overline{OE}$
T_3	140		ns	$\overline{OE}$ PULSE WIDTH (3xDMACLK) −10 nsec
T_4	25		ns	READ DATA SETUP TO $\overline{OE}$
T_5		35	ns	$\overline{OE}$ TO DATA HIGH IMPEDANCE (DMACLK) −15 nsec
T_6		25	ns	$\overline{WE}$ TO WRITE DATA VALID
T_8	100		ns	$\overline{OE}$ TO TRAILING EDGE OF $\overline{DACK}$ (2.5xDMACLK) −25 nsec
T_9	15		ns	TRAILING EDGE OF $\overline{DACK}$ TO TRAILING EDGE OF $\overline{OE}$ (0.5xDMACLK) −10 nsec
T_{10}	0		ns	ADDR SETUP TIME TO $\overline{WE}$
T_{11}	25		ns	WRITE RECOVERY TIME (ADDR HOLD AFTER $\overline{WE}$ INACTIVE) (DMACLK) −25 nsec

SYMBOL	MIN	MAX	UNITS	COMMENTS
T_{12}	85		ns	$\overline{WE}$ PULSE WIDTH (2xDMACLK) −15 nsec
T_{13}	85		ns	WRITE DATA SETUP TIME (2xDMACLK) −15 nsec
T_{14}	25		ns	WRITE DATA HOLD TIME (DMACLK) −25 nsec
T_{15}	85		ns	$\overline{DACK}$ TO TRAILING EDGE OF $\overline{WE}$ (2xDMACLK) −15 nsec
T_{16}	10		ns	TRAILING EDGE OF $\overline{WE}$ TO TRAILING EDGE OF $\overline{DACK}$ (0.5xDMACLK) −15 nsec
T_{17}	50		ns	$\overline{DACK}$ PULSE WIDTH HIGH (1.5xDMACLK) −25 nsec
T_{18}	100		ns	$\overline{DACK}$ PULSE WIDTH LOW (2.5xDMACLK) −25 nsec
T_{1a}	85		ns	S0, S1 HOLD AFTER $\overline{OE}$ ACTIVE (2xDMACLK) −15 nsec
T_{1b}	85		ns	S0, S1 HOLD AFTER $\overline{WE}$ ACTIVE (2xDMACLK) −15 nsec
T_{1c}	35	65	ns	S0, S1 VALID TO ADDRESS VALID (DMACLK) −15 nsec (MIN) (DMACLK) +15 nsec (MAX)
T_{1d}	185		ns	S0, S1 WIDTH (4xDMACLK) −15 nsec
T_{2a}	135		ns	$\overline{WRITE}$ HOLD AFTER $\overline{OE}$ ACTIVE (3xDMACLK) −15 nsec
T_{2b}	135		ns	$\overline{WRITE}$ HOLD AFTER $\overline{WE}$ ACTIVE (3xDMACLK) −15 nsec
T_{2c}	−15	15	ns	$\overline{WRITE}$ VALID TO ADDRESS VALID
T_{2d}	185		ns	$\overline{WRITE}$ WIDTH (4xDMACLK) −15 nsec

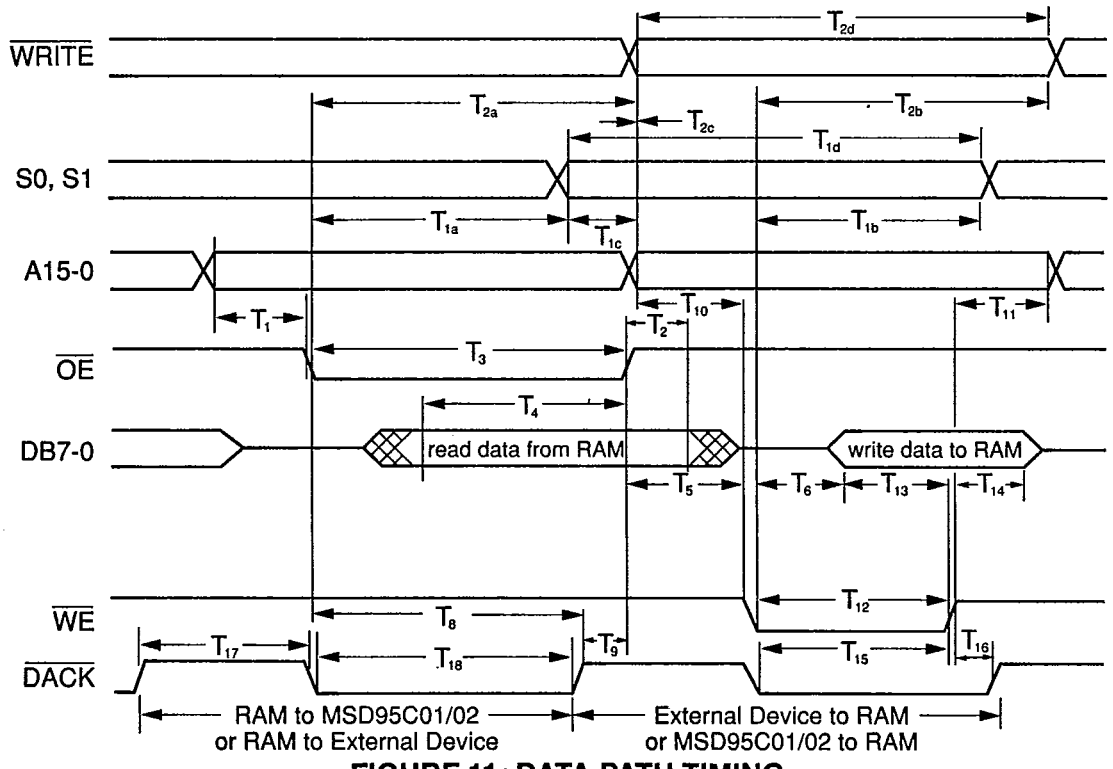


FIGURE 11: DATA PATH TIMING

T-52-33-63

DMA CLOCK and DMA REQUEST TIMING

SYMBOL	MIN	MAX	UNITS	COMMENTS
T_{20}	50	1000	nsec	DMACLK CYCLE TIME
T_{21}	20		nsec	DMACLK HIGH TIME
T_{22}	20		nsec	DMACLK LOW TIME
T_{23}		5	nsec	DMACLK RISE TIME
T_{24}		5	nsec	DMACLK FALL TIME
T_{25}	1.1xT20		nsec	$\overline{DREQ}$ CYCLE TIME (Note 1)
T_{26}	20		nsec	$\overline{DREQ}$ LOW TIME
T_{27}	20		nsec	$\overline{DREQ}$ HIGH TIME
T_{28}		5	nsec	$\overline{DREQ}$ RISE TIME
T_{29}		5	nsec	$\overline{DREQ}$ FALL TIME

NOTE 1: Can be 1XT21 if $\overline{DREQ}$ is synchronized to DMACLK.

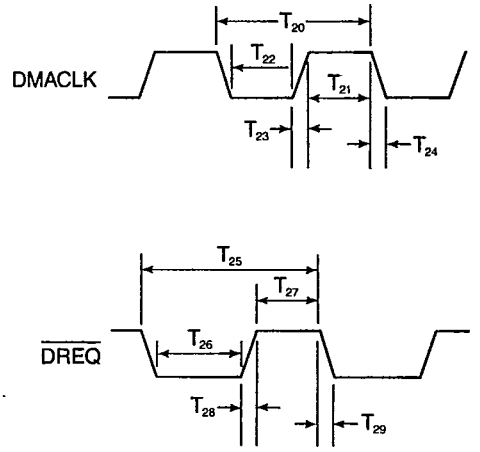


FIGURE 12: DMA CLOCK and DMA REQUEST TIMING

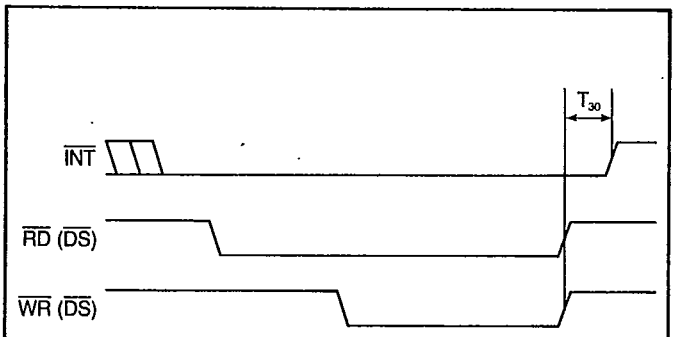


FIGURE 13: INTERRUPT TIMING

INTERRUPT TIMING

SYMBOL	MIN	MAX	UNITS	COMMENTS
T_{30}		100	nsec	INTERRUPT RESET FROM $\overline{RD}$ or $\overline{WR}$

DISK (Write Data) TIMING

SYMBOL	MIN	MAX	UNITS	COMMENTS
T_1	48		nsec	CYCLE TIME
T_2	0.40 T1		nsec	HIGH TIME
T_3	0.40 T1		nsec	LOW TIME
T_4		5	nsec	RISE TIME
T_5		5	nsec	FALL TIME
T_6	(TBD)		nsec	DATA SETUP TIME
T_7	(TBD)		nsec	DATA HOLD TIME
T_8	0.71 T1	1.25 T1	nsec	CYCLE TIME
T_9		10	nsec	WDATA VALID TO EARLY, LATE
T_{9a}		10	nsec	EARLY, LATE VALID TO WDATA VALID
T_{10}	0.5 T1		nsec	Hold from trailing edge of WDATA
T_{11}	2xT1-20		nsec	Active Time

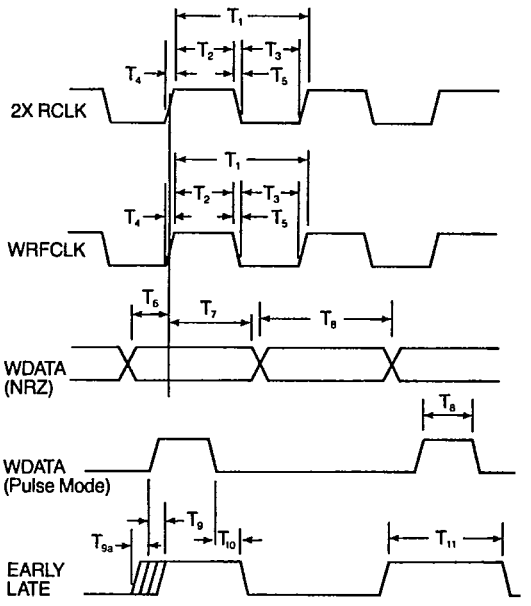


FIGURE 14: DISK (Write Data) TIMING

Microprocessor Interface Timing

T-52-33-63

SYMBOL	R/$\overline{W}$ $\overline{DS}$	$\overline{RD}$ $\overline{WR}$	UNITS	COMMENTS
T_1	45	45	nsec min	ALE ($\overline{AS}$) ACTIVE PULSE WIDTH
T_2	30	30	nsec min	ADDRESS VALID TO ALE INACTIVE
T_3	15	15	nsec min	ALE INACTIVE TO ADDRESS INVALID
T_4	120	120	nsec min	READ STROBE LOW PULSE WIDTH
T_5	70	115	nsec MAX	$\overline{RD}$ ($\overline{DS}$) ACTIVE TO READ DATA VALID
T_6	10	10	nsec min	READ DATA HOLD FROM $\overline{RD}$ ($\overline{DS}$) HIGH
T_7	110	110	nsec min	WRITE STROBE LOW PULSE WIDTH
T_8	135	135	nsec min	WRITE DATA VALID TO $\overline{WR}$ ($\overline{DS}$) INACTIVE
T_9	25	25	nsec min	WRITE DATA HOLD FROM $\overline{WR}$ ($\overline{DS}$) INACTIVE
T_{10}	20	20	nsec min	ALE ($\overline{AS}$) INACTIVE TO READ STROBE ACTIVE
T_{11}	20	20	nsec min	ALE ($\overline{AS}$) INACTIVE TO WRITE STROBE ACTIVE
T_{12}	0	0	nsec min	$\overline{RD}$ ($\overline{DS}$) INACTIVE TO ALE ACTIVE
T_{13}	0	0	nsec min	$\overline{WR}$ ($\overline{DS}$) INACTIVE TO ALE ACTIVE
T_{14}	0	0	nsec MAX	$\overline{CE}$ VALID TO $\overline{RD}$, $\overline{WR}$ OR $\overline{DS}$
T_{15}	0	0	nsec min	R/$\overline{W}$ VALID TO ALE ($\overline{AS}$) INACTIVE

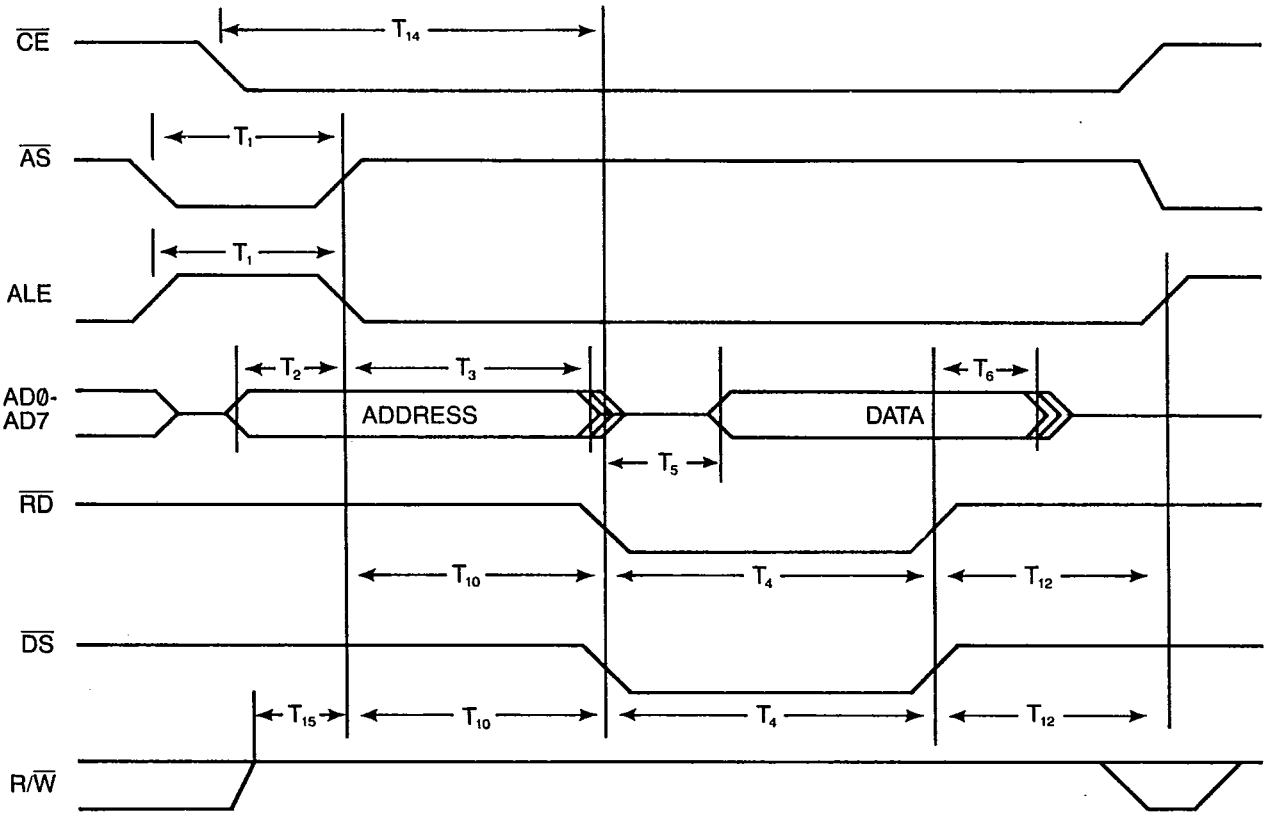


FIGURE 15: MICROPROCESSOR INTERFACE TIMING (READ)

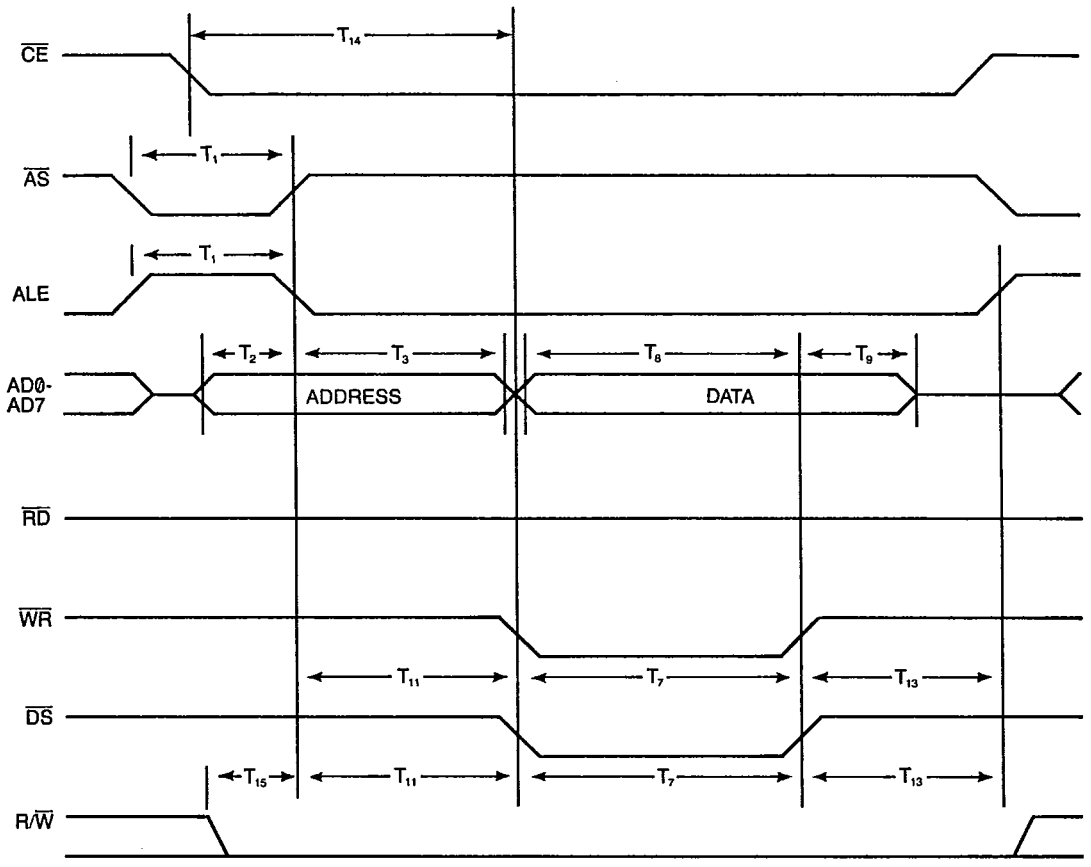


FIGURE 16: MICROPROCESSOR INTERFACE TIMING (WRITE)

DISK INTERFACE TIMING

SYMBOL	MIN	MAX	UNITS	COMMENTS
T_{40}	(TBD)		nsec	Write gate setup to AMENA
T_{40a}	8xWCLK		nsec	Write gate active time
T_{41}	(TBD)		nsec	Write gate hold after AMENA
T_{42}	8xWCLK		nsec	AMENA active time
T_{43}	50		nsec	Index active time
T_{44}	(TBD)		nsec	AMENA TO AMFND
T_{45}	(TBD)		nsec	AMENA inactive to AMFND inactive
T_{46}	50		nsec	AMFND active time
T_{47}	(TBD)		nsec	AMFND active to RGATE active
T_{49}	25		nsec	$\overline{\text{RDATA}}$ active time
T_{50}	12		nsec	$\overline{\text{RDATA}}$ setup to 2xRCLK
T_{51}	12		nsec	$\overline{\text{RDATA}}$ hold from 2xRCLK
T_{52}	25		nsec	SNS1 active time
T_{53}	25		nsec	TST3 active time
T_{54}	25		nsec	SNS2 active time
T_{55}	25		nsec	SNS3 active time
T_{56}	25		nsec	SNS4 active time
T_{57}	25		nsec	TST2 active time
T_{58}	1		µsec	$\overline{\text{RST}}$ active time
T_{59}	0		nsec	$\overline{\text{LPS}}$ active time (Note 1)
T_{60a}	8xWCLK		nsec	RGATE active time

NOTE 1

If the chip is at idle, then the time from $\overline{\text{LPS}}$ becoming active until the chip actually enters the low power mode is the longest of the following times:

1. (2 x (2XRCLK)) +200nsec
2. (2 x (WRFCLK)) +200nsec
3. (0.5 x (DMACLK)) +200nsec

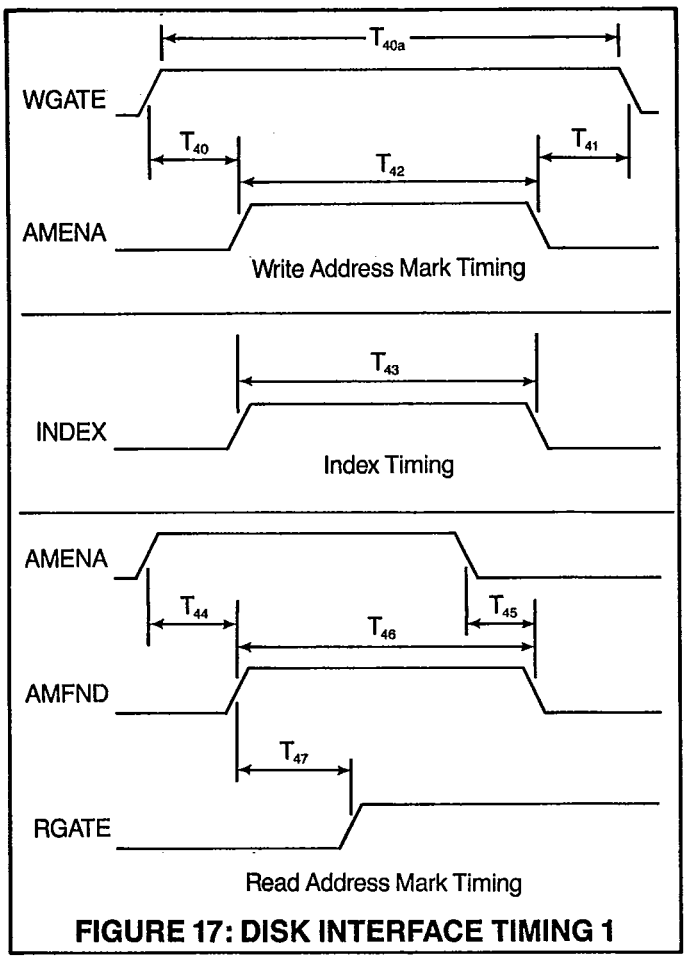


FIGURE 17: DISK INTERFACE TIMING 1

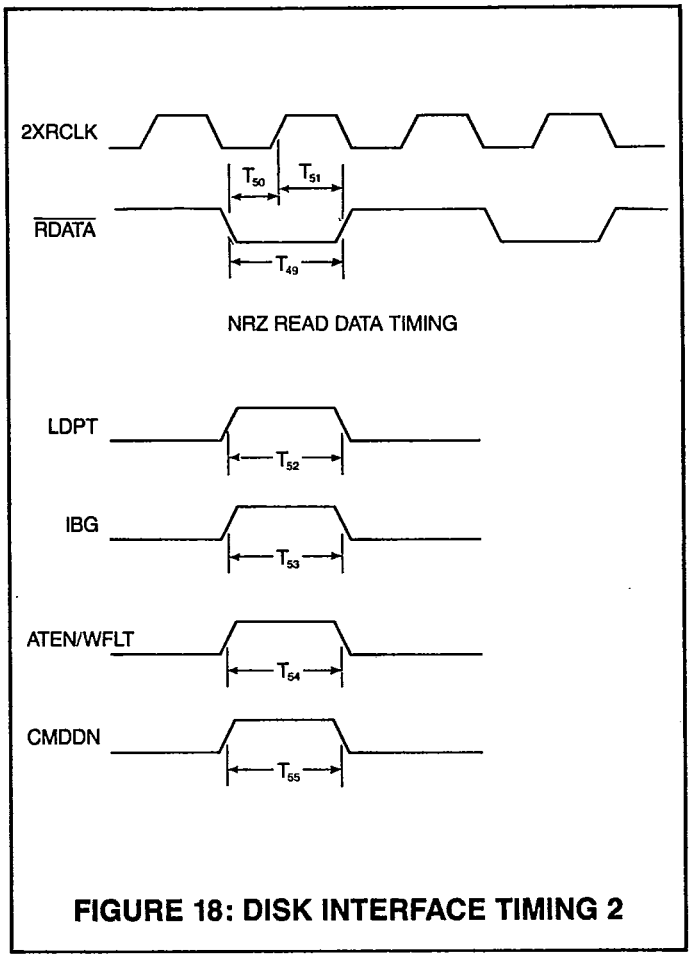


FIGURE 18: DISK INTERFACE TIMING 2

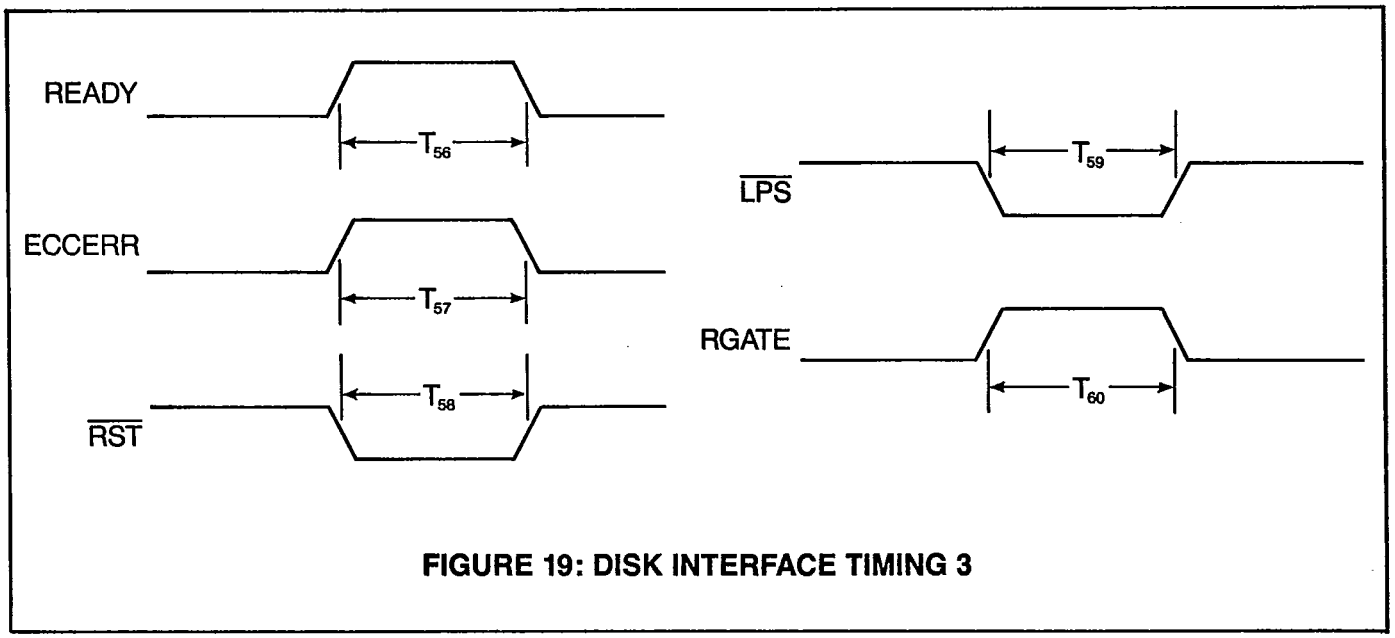


FIGURE 19: DISK INTERFACE TIMING 3

Unix® is a registered trademark of AT&T Bell Labs.
SuperCell™ is a trademark of Standard Microsystems Corporation.

IBM® and AT® are registered trademarks of the International Business Machines Corporation.

PC/XT™ is a trademark of International Business Machines Corporation.

STANDARD MICROSYSTEMS CORPORATION
35 Marcus Blvd. Hauppauge, N.Y. 11788
(516) 273-3100 • FAX 516-231-6004