WD10C01A

Winchester

Disk Controller

WESTERN DIGITAL

TABLE OF CONTENTS

Section	Title	Page
1.0	INTRODUCTION	20-1
	1.1 Features	20-1
2.0	GENERAL DESCRIPTION	20-2
3.0	SYSTEM BLOCK DIAGRAM	20-3
4.0	SIGNAL DESCRIPTION	20-4
5.0	ARCHITECTURE	20-7
	5.1 Error Correction And Detection Codes	20-8
	5.1.1 CCITT-CRC	20-8
	5.1.2 Reed-Solomon ECC	20-8
6.0	PROGRAMMING REFERENCE	20-9
	6.1 Register Assignments	20-9
	6.2 Register Definitions	20-10
	6.3 Configuration Group	20-11
	6.3.1 SRESET - Set Hardware Reset Register (00)	20-11
	6.3.2 SISR - Interrupt Status Register (01)	20-12
	6.3.3 SIMR - Interrupt Mask Register (02)	20-13
	6.3.4 SEQSTS - Sequencer Status (03)	20-14
	6.3.5 PYC - Port Y Configuration (03)	20-15
	6.3.6 Control Store Windows	20-16
	6.3.6.1 CSERR - Control Store Error Control Byte Window (04)	20-16
	6.3.6.2 CSCTL - Control Store Control Byte Window (05)	20-17
	6.3.6.3 CSVAL - Control Store Value Byte Window (06)	20-18
	6.3.6.4 CSCNT - Control Store Count Byte Window (07)	20-20
	6.3.6.5 Wait Condition Sequences	20-21
	6.4 Device Control Group	20-22
	6.4.1 PORTX - PORTX Output Bits (08)	20-22
	6.4.2 PORTY - PORT Y I/O Bits (09)	20-22
	6.4.3 PORTZ - PORT Z Input Bits (0A)	20-22
	6.4.4 AMC - Address Mark Control (0A)	20-23
	6.4.5 SEQCTL - Sequencer Control Register (0B)	20-24
	6.4.6 START - Sequencer Start Address (0C)	20-25
	6.4.7 LOOP - Sequencer Loop Address (0D)	20-26
	6.4.8 ECCCTL - Error Correction Control Register (0E)	20-27
	6.4.9 SECCNT - Sector Count Register (0F)	20-28
	6.4.10 ECCP- ECC Parameter Register (10)	20-29
	6.4.11 ECCS - RS-ECC Status Register (11)	20-30

20

Section	Title	Page
	6.4.12 SPORT - Syndrome Port (12)	20-31
	6.4.13 TEST - Test Register (16)	20-31
	6.4.14 SKIP - Skip Address Register (17)	20-32
	6.5 ID Register Group	20-33
	6.6 Device Programming	20-35
	6.6.1 Initialization	20-35
	6.6.2 Command Programming	20-35
	6.6.3 Control Store Programming	20-35
	6.6.4 Programming Examples	20-37
	6.6.4.1 Format Track Example	20-38
	6.6.4.2 Read Sector Example	20-40
	6.6.4.3 Write Sector Example	20-42
	6.6.4.4 Read And Write Sector Example	20-44
	6.6.5 ID Retry And Error Conditions	20-46
	6.6.6 Error Recovery	20-47
	6.6.7 Error Correction	20-48
7.0	DC ELECTRICAL SPECIFICATIONS	20-49
	7.1 Absolute Maximum Ratings	20-49
	7.2 Standard Test Conditions	20-49
	7.3 DC Characteristics	20-50
8.0	AC OPERATING CHARACTERISTICS	20-51
	8.1 OSC And CPUCLK Timing	20-51
	8.2 CPU Interface Timing	20-54
	8.3 Buffer Interface Timing	20-58
	8.4 Serial Data Timing	20-61
A.0	WD10C01A PROGRAMMER'S BENCH REFERENCE (PBR)	20-63
	A.1 Address Bit Tables	20-63
B.0	RESET CONDITIONS	20-66
C.0	CRYSTAL OSCILLATOR APPLICATIONS	20-67
D.0	PIN/SIGNAL SUMMARY	20-68
E.0	DIFFERENCES BETWEEN WD10C00 AND WD10C01A	20-69
	E.1 Error Correction And Detection	20-69
	E.2 SRESET Register	20-69
	E.3 ECCCTL Register	20-69
	E.4 Timing	20-69
	E.5 Pin Name	20-69
	E.6 Parity Error Handling	20-69
	E.7 SEQCTL Register	20-69

LIST OF ILLUSTRATIONS

Figure	Title	Page
1-1	WD10C01A Pin Diagram	20-1
3-1	System Block Diagram	20-3
8-1	TTL Source X1 Clock Input	20-51
8-2	OSC Output	20-51
8-3	CPUCLK Output	20-52
8-4	Microprocessor RD* Timing (RD* Controlled)	20-54
8-5	Microprocessor RD* Timing (CS* Controlled)	20-54
8-6	Microprocessor WR* Timing (WR* Controlled)	20-55
8-7	Microprocessor WR* Timing (CS* Controlled)	20-55
8-8	Reset Timing	20-56
8-9	Externally Generated Interrupt Timing	20-56
8-10	Asynchronous Mode Data Bus Read Timing (1)	20-58
8-11	Asynchronous Mode Data Bus Read Timing (0)	20-58
8-12	Asynchronous Mode Data Bus Write Timing (1)	20-59
8-13	Asynchronous Mode Data Bus Write Timing (0)	20-59
8-14	NRZ Data Input Timing	20-61
8-15	NRZ Data Output Timing	20-61
C-1	Oscillator with Capacitors	20-67

LIST OF TABLES

Table	Title	Page
4-1	CPU Interface	20-4
4-2	Buffer Interface	20-5
4-3	Disk Data Interface	20-5
4-4	Disk Control Interface	20-6
4-5	Device Power	20-6
6-1	Register Groups	20-9
6-2	SRESET- Set Hardware Register (00)	20-11
6-2A	ID Register Selection	20-12
6-3	SISR - Interrupt Status Register	20-12
6-4	SIMR - Interrupt Mask Register	20-13
6-5	SEQSTS - Sequencer Status Register	20-14
6-6	PYC - Port Y Configuration (03)	20-15
6-7	CSERR - Control Store Error Control Byte Window	20-16
6-8	CSCTL - Control Store Byte Window (05)	20-17
6-9	SVSEL of Control Byte Zero	20-18
6-10	SVSEL of Control Byte One	20-19
6-11	CWSEL of Control Byte Zero	20-20
6-12	CWSEL of Control Byte One	20-20

Table	Title	Page
6-13	PortX Output Bits (08)	20-22
6-14	PortY I/O Bits (09)	20-22
6-15	PortZ Input Bits (0A)	20-22
6-16	AMC - Address Mark Control (0A)	20-23
6-17	SEQCTL - Sequencer Control Register (0B)	20-24
6-18	START - Sequencer Start Address (0C)	20-25
6-19	LOOP - Sequencer Loop Address (0D)	20-26
6-20	ECCCTL - Error Correction Register (1E)	20-27
6-21	SECCNT - Sector Count Register (0F)	20-28
6-22	ECCP - ECC Parameter Register (10)	20-29
6-23	ECCS - RS-ECC Status Register (11)	20-30
6-24	SPORT - Syndrome Port (12)	20-31
6-25	SKIP - Skip Address Register (17)	20-32
6-26	Skip Control Store Example	20-32
6-27	ID0 - ID Register 0 (18)	20-33
6-28	ID1 - ID Register 1 (19)	20-33
6-29	ID2 - ID Register 2 (1A)	20-33
6-30	ID3 - ID Register 3 (1B)	20-33
6-31	ID4 - ID Register 4 (1C)	20-34
6-32	ID5 - ID Register 5 (1D)	20-34
6-33	ID6 - ID Register 6 (1E)	20-34
6-34	ID7 - ID Register 7 (1F)	20-34
6-35	Write Byte Sync Example	20-36
6-36	Data Field Size Example	20-36
6-37	Format Track Example Control Store	20-38
6-38	Read Sector Example Control Store	20-40
6-39	Write Sector Example Control Store	20-42
6-40	Read and Write Sector Example Control Store	20-44
7-1	DC Characteristics	20-50
8-1	OSC and CPUCLK Timing Parameters	20-53
8-2	CPU Interface Timing Parameters	20-57
8-3	Buffer Interface Timing Parameters	20-60
8-4	Serial Data Timing Parameters	20-62

1.0 INTRODUCTION

The WD10C01A is a VLSI Winchester/Optical Disk Controller chip that provides the data handling and control for intelligent disk applications. The WD10C01A interfaces to nearly any serial disk interface, including ST412, ST412HP, ESDI, SMD, and many optical disk interfaces. The WD10C01A provides great flexibility in format design, allowing for multiple ID fields, special synchronization requirements, special information fields, or almost any other special requirement. The WD10C01A can provide all of the data, status, and control signals required by these interfaces.

1.1 FEATURES

- Disk interfaces and formats supported include ST412, ST412HP, ESDI, SMD and optical disks
- Full multi-sector operation with four byte ID auto-increment
- Up to 24 mbit/second maximum transfer rate
- Supports 16-bit CRC-CCITT polynomial on ID field
- Degree 5 and 6 Reed-Solomon ECC with 3- or 5-way interleave to protect data field against long error burst
- Provides composite syndromes for error correction
- Up to 1:1 interleave operation
- Writeable control store allows flexible error Recovery, including redundant ID and sync fields

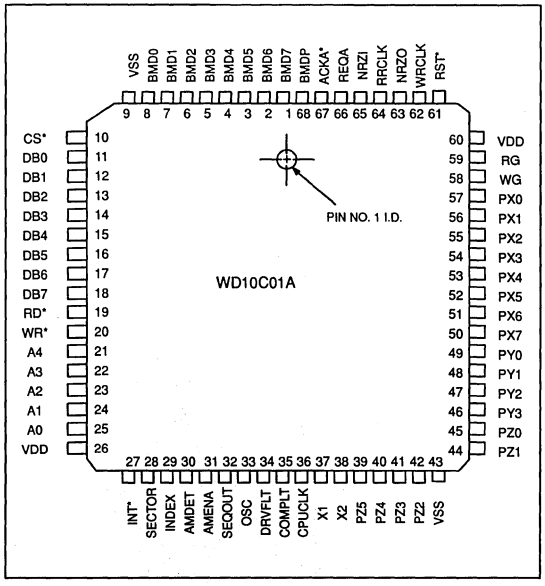

FIGURE 1-1. WD10C01A PIN DIAGRAM

- Support hard or soft sector formats, sector size to 1250 bytes when 5-way interleave is used for degree 5 Reed-Solomon code
- Built in crystal driver for data rate and/or CPU use
- Generic non-multiplexed CPU interface with maskable interrupts
- Separate CPU and disk data busses
- On-the-fly compare against buffer data
- 20 general I/O Lines for disk drive control
- 68-Pin PLCC package

2.0 GENERAL DESCRIPTION

The WD10C01A has separate ports for data DMA transfer and for the microprocessor to achieve a maximum performance.

The WD10C01A performs the disk data serialization and de-serialization. It can interface with various magnetic and optical Data Encoder/Decoders (ENDEC). The data format on the disk is controlled by a Writeable Control Store. It is very flexible with the capability to support various formats including optical disk. The device also has the capability to compare data and verify the ECC. The WD10C01A can perform full track operations without CPU intervention using the Writeable Control Store, auto-incrementing ID registers and the sector counter.

The WD10C01A includes logic implementing CCITT-CRC and Reed-Solomon ECC for data protection. The ID field is protected by sixteen bit CRC and the data field is protected by degree five or six RS-ECC. The user can also select the interleave factor of three or five for the data field. The term "interleave" here should not be confused with the term "sector interleave," which defines the relation between the physical and the logical location of sectors within a track. Interleaving the data field means spreading the data across several ECC code words to improve the capability for correcting longer error bursts.

For the error correction, the WD10C01A generates the composite syndromes. From this error information, the correction software can generate individual syndromes to correct up to two error bytes per interleave (for degree 5) or up to three error bytes per interleave (for degree 6). Optionally, the user can use external, more powerful ECC device, such as WD60C80.

The highly programmable nature of the WD10C01A allows the use of redundant ID and data sync fields within a single sector. This feature, along with the programmable degree 5 or 6 RS-ECC, gives the WD10C01A a greater capability for recovering user data in a sector with 'grown' defects.

The WD10C01A interfaces to the buffer manager, such as WD60C40 through an eight bit DMA port. It uses asynchronous protocol through DREQ/$\overline{\text{DACK}}$ signals.

The WD10C01A has a generic microprocessor interface that allows the WD10C01A to be used with all popular 8-bit microprocessors. The WD10C01A has interrupt capability, which frees up the microprocessor from constantly polling the device status. The WD10C01A also has a built-in crystal oscillator driver that can be used to generate data reference, buffer management, or microprocessor clocks. Two separate outputs are provided with internal programmable dividers. Both outputs have the extra drive voltage and current necessary for driving MOS microprocessor clock inputs.

The WD10C01A has 20 lines dedicated to external I/O ports that the microprocessor can use to control the drive and head select lines, seek command and drive status. Eight lines are output only, six lines are input only, and four other lines can be individually programmed for input or output. Two other latch and hold input lines are tied to the interrupt logic and can be used to detect fault and ready conditions without constantly polling the device.

3.0 SYSTEM BLOCK DIAGRAM

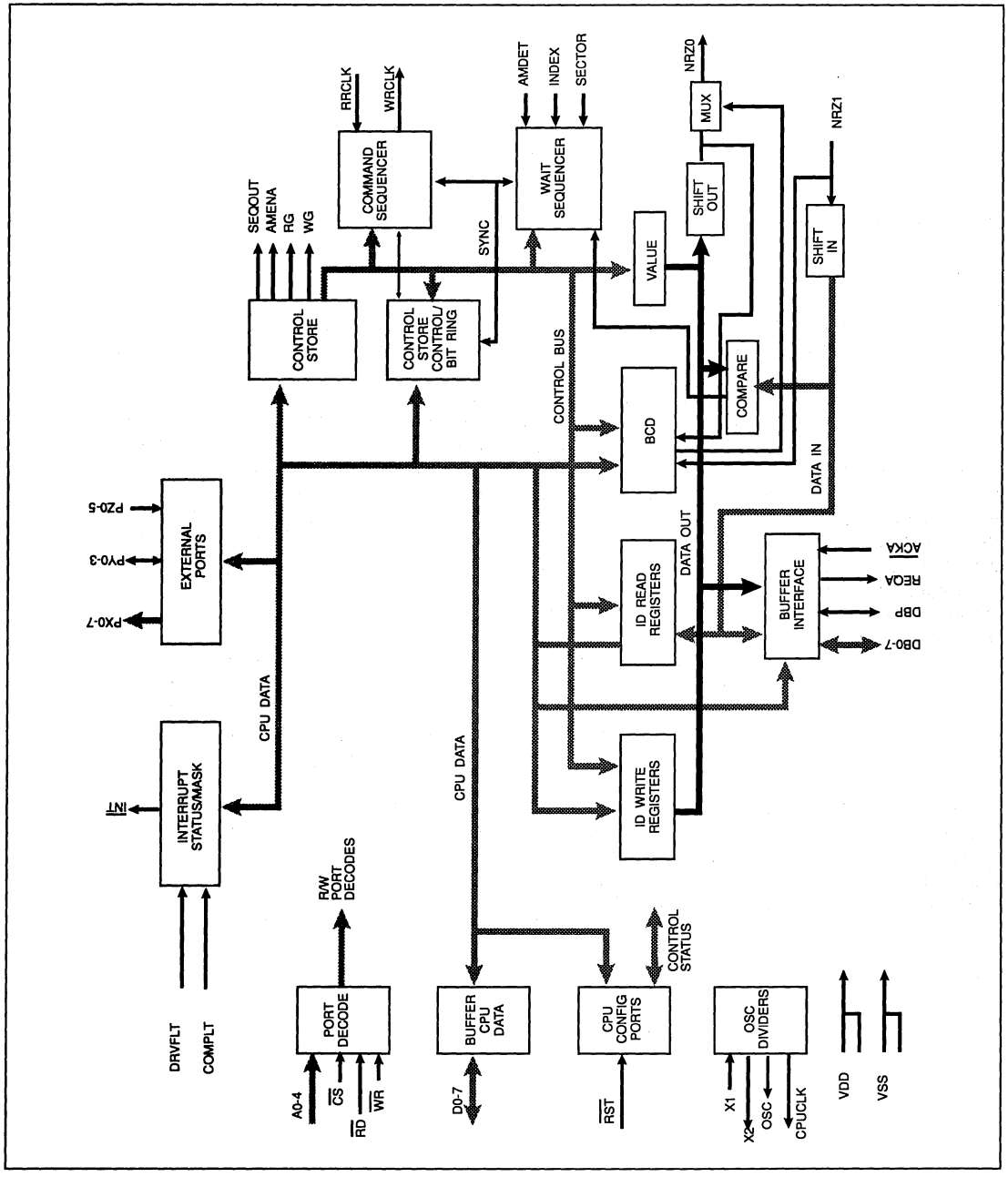

FIGURE 3-1. SYSTEM BLOCK DIAGRAM

4.0 SIGNAL DESCRIPTION

The WD10C01A is a 68-pin PLCC device.

The following section describes the external signals available on the WD10C01A. Conventions are as follows:

I indicates that a signal is an input to the WD10C01A.

O indicates that a signal is an output from the WD10C01A.

I/O indicates that a signal is bidirectional.

I,O indicates that a signal can be input output.

* as a suffix indicates an active low signal; however, most drawings and text use an overscore to indicate an active low signal.

PIN	MNEMONIC	I/O	DESCRIPTION
25-21	A0-A4	I	CPU ADDRESS BUS. These signals are used to address internal WD10C01A registers.
36	CPUCLK	O	CPU CLOCK OUTPUT. This is the OSC output (see below) divided by two or three, selected by an internal register. This output has extra drive for use with certain microprocessors.
10	$\overline{CS}$	I	CHIP SELECT. This active low signal enables the WD10C01A bus interface logic.
11-18	DB0-DB7	I/O	CPU DATA I/O BUS. This data bus is used to transfer data between the CPU and the WD10C01A.
27	$\overline{INT}$	O	CPU INTERRUPT. This active low, open drain output is asserted whenever an enabled interrupt condition occurs on the WD10C01A.
33	OSC	O	OSCILLATOR OUTPUT. The 1x crystal oscillator output, optionally divided by two. This signal has the same drive capability as CPUCLK.
19	$\overline{RD}$	I	CPU READ STROBE. This active low signal enables data from the WD10C01A on to the CPU data bus.
61	$\overline{RST}$	I	RESET. This active low signal resets all internal circuits that must be reset at power on. A complete list is given later in this document. The reset is latched and the condition must be cleared by the CPU.
20	$\overline{WR}$	I	CPU WRITE STROBE. This active low signal strobes data into the selected WD10C01A register from the CPU data bus.
37	X1	I	CRYSTAL DRIVER INPUT. X1 can also be driven by an external clock.
38	X2	O	CRYSTAL DRIVER OUTPUT.

TABLE 4-1. CPU INTERFACE

PIN	MNEMONIC	I/O	DESCRIPTION
67	$\overline{\text{ACKA}}$	I	BUFFER DATA ACKNOWLEDGE. This active low signal indicates to the WD10C01A that data can now be transferred to or from the data buffer.
8-1	BMD0-7	I/O	BUFFER MEMORY DATA BUS. This is an eight bit data bus that interfaces the WD10C01A with the disk data buffer memory.
68	BMDP	I/O	DATA BUS PARITY. This signal is used to generate and check parity with the disk data buffer memory.
66	REQA	O	BUFFER DATA REQUEST. This signal is asserted when the WD10C01A has data to write to the data buffer, or needs data from the data buffer.

TABLE 4-2. BUFFER INTERFACE

PIN	MNEMONIC	I/O	DESCRIPTION
30	AMDET	I	ADDRESS MARK DETECTED. Used only in ST412 type interfaces that use missing clocks or other qualifiers to the sync bytes that mark the start of a field.
31	AMENA	O	ADDRESS MARK ENABLE. Used to write a missing clock sync byte (ST412) or soft sector mark (ESDI, SMD) on the media.
65	NRZI	I	NRZ READ DATA IN. Serial data input from the disk phase-locked loop. This signal is clocked in by the rising edge of RRCLK.
63	NRZO	O	NRZ WRITE DATA OUT. Serial data output. NRZO is valid on the rising edge of WRCLK.
59	RG	O	READ GATE. Active when reading from the disk drive. This signal is turned off for one byte time on an ID search error to reset external data decoders.
64	RRCLK	I	READ/REFERENCE CLOCK. This is the reference clock used to set the data rate for write, and is the recovered clock for read. The switching must be glitch free. NRZI is clocked into the WD10C01A by the rising edge of this clock.
32	SEQOUT	O	SEQUENCER OUTPUT. This signal is a user definable output bit that is set up in the control byte of the sequencer control store (see below). This signal can be used to control an external ECC generator and checker, and is byte aligned with both read and write data.
58	WG	O	WRITE GATE. Active when writing to the disk drive.
62	WRCLK	O	WRITE CLOCK. This is output during write for drives that require it. NRZO data is valid on the rising edge of this clock.

TABLE 4-3. DISK DATA INTERFACE

PIN	MNEMONIC	I/O	DESCRIPTION
35	COMPLT	I	COMPLETE. This signal is used to detect function complete conditions, such as seeks or status requests. The signal only generates a CPU interrupt, and does not interfere with a read or write operation.
34	DRVFLT	I	DRIVE FAULT. This signal is used to detect faults from the drive. The signal only generates a CPU interrupt, and does not interfere with a read or write operation.
29	INDEX	I	INDEX. This signal is used to indicate the start of a track. This signal is latched for CPU status and interrupt.
57-50	PX0-7	O	PORT X. This general purpose output port is intended for use as drive select and head select signals.
49-46	PY0-3	I,O	PORT Y. This general purpose port is intended for use as other control outputs or inputs. Each bit is selectable as input or output, but all bits are initialized to input when the WD10C01A is reset.
45,44, 42-39	PZ0-5	I	PORT Z. This general purpose input port is used to receive drive status signals.
28	SECTOR	I	SECTOR MARK. This signal is used for marking sector start locations on the media. This can either be a hard sector mark, or a soft mark written on the media using AMENA (ESDI or SMD).

TABLE 4-4. DISK CONTROL INTERFACE

PIN	MNEMONIC	I/O	DESCRIPTION
26,60	VDD	I	+5 VOLTS DC.
9,43	VSS	I	GROUND.

TABLE 4-5. DEVICE POWER

5.0 ARCHITECTURE

The WD10C01A consists of the functional blocks shown in the block diagram in Figure 3-1. VDD and VSS are applied to the device through two separate pins each to improve noise immunity. The top and left hand sides of the diagram show CPU interface features, the right hand side shows disk interface features, and the bottom shows buffer interface features. These blocks are discussed in the following paragraphs.

The PORT DECODE block generates the 28 write strobes and 25 read strobes used by the microprocessor to access the various internal control and status ports. These include the interrupt registers, external disk control ports, control store, control store control, configuration, ECC control, and ID registers.

The BUFFER CPU DATA block controls the transfer of data between the microprocessor and the internal registers. The direction control is qualified by chip select ($\overline{CS}$) and read strobe ($\overline{RD}$).

The CPU CONFIGURATION PORTS are used to reset the WD10C01A, set the address mark enable timing, set the buffer interface timing, and select the frequency of the clock outputs, OSC and CPUCLK.

The OSCILLATORS AND DIVIDERS block generates the clock outputs, OSC and CPUCLK, using an external crystal (or clock input) and dividers to select the frequency. Frequency selection is glitch free.

The INTERRUPT STATUS AND MASK registers are used to check and mask interrupts. The mask register does not affect the status register inputs. The interrupt sources include index and sector mark, drive fault and operation complete, ECC errors, and internal event status.

The EXTERNAL PORTS are used to generate control signals and read status with the disk drive. Eight bits are output, six are input, and four are individually programmable for either input or output.

The CONTROL STORE consists of 32 words of 28 bits that are used to program the format of the disk sector. The data source, field length, error handling and checksum selection, and control signals, like Read Gate and Write Gate, are controlled by the data stored here.

The CONTROL STORE CONTROL determines the next address in the control store to use, whether the next sequential address or a jump to another address. This block includes the sector counter used for multi-sector commands. This block also includes the BIT RING COUNTER, which determines the timing of data transfers in the WD10C01A.

The WAIT SEQUENCER handles searches for index, sector mark, address mark, and byte synchronization.

The ID WRITE REGISTERS are 8 eight bit registers that are used to set the ID write field for format, or the search field for read/update write. Four of the registers are counters that auto-increment during multi-sector commands. The other four registers do not increment, and are used for defect and flag information. The first byte of the four counters can be disabled for three byte ID fields.

The ID READ REGISTERS are used to read the last ID read from the media to aid in defect handling.

The ECD block performs the CRC on ID, selects the degree of RS-ECC with 3- or 5-way interleave on data fields, generates the checksum bytes, creates the composite syndromes and ECC error status necessary for the calculation of error location and mask.

The VALUE register holds immediate data from the control store when generating gaps, sync fields, and address mark bytes.

The SHIFT OUT register serializes internal or external (buffer) data for writing on the disk. The output is multiplexed with the output of the checksum register.

The SHIFT IN register de-serializes the read data from the disk, clocked in by RRCLK. The data is also transferred to the checksum register for checking.

The COMPARE block is used to compare incoming read data with an internal or external data

source. These include byte synchronization detection, ID field search, and buffer data compare.

The BUFFER INTERFACE handles the fetching and writing of data with the external data buffer. This includes parity generation and checking, and data handshake with the buffer controller.

5.1 ERROR CORRECTION AND DETECTION CODES

5.1.1 CCITT-CRC

The WD10C01A protects the ID fields using the CCITT-CRC code. The polynomial is defined as follow:

$$g(x) = x^{16} + x^{12} + x^5 + 1$$

The user can select the initial state of the shift registers to be either all zero's or all one's.

5.1.2 Reed-Solomon ECC

The data fields are protected using the interleaved Reed-Solomon code operating on 8 bits symbols. The redundancy bytes are inverted. WD10C01A supports two polynomials of degree five and six. The generator polynomials operate in the finite field GF(256), which are defined as follow:

Let β^i represent elements of a finite field defined by a polynomial over GF(2):

$$g(x) = x^8 + x^5 + x^3 + x^2 + 1.$$

The elements of the finite field employed by the codes are:

$$a^i = (\beta^i)^{88}.$$

The generator polynomials are self reciprocal and defined as follows:

1. Degree 5, distance 6:

$$G(x) = (x+a^{253})*(x+a^{254})*(x+a^0)*(x+a^1)*(x+a^2).$$

The coefficients of the polynomial in decimals are:

1, 60, 183, 183, 60, 1.

2. Degree 6, distance 7:

$$G(x) = (x+a^{125})*(x+a^{126})*(x+a^{127})*(x+a^{128})*$$
$$(x+a^{129})*(x+a^{130}).$$

The coefficients of the polynomial in decimals are:

1, 176, 126, 163, 126, 176, 1.

The calculated checkbytes are **inverted** before they are written into the disk. During read operation, the WD10C01A computes the composite syndromes by recalculating the checkbytes for the data field being read and compare them against the checkbytes written on the disk. The individual syndromes can be computed by dividing the composite syndromes with the factors of the generator polynomial:

$$S_i(x) = R(x) \text{ MODULO } G_i(x)$$

where:

i = 0..ecc degree -1.

$S_i(x)$ = i'th individual syndrome.

R(x) = composite syndrome.

$G_i(x)$ = factor of the generator polynomial, $(x+a^i)$.

The WD10C01A supports two data interleaving factors. The user can optimize the performance by choosing the right degree and interleave combination.

The following table shows the redundancy overhead and maximum field size.

Interleave Factor	Degree 5		Degree 6	
	Overhead*	Max. Data* Field Size*	Overhead*	Max. Data* Field Size*
3	15	750	18	747
5	25	1250	30	1245

* all units bytes

6.0 PROGRAMMING REFERENCE

6.1 REGISTER ASSIGNMENTS

The WD10C01A contains 32 output and 23 input ports distributed in a 32 port address space, selected by A0-A4, $\overline{CS}$, and $\overline{RD}$ or $\overline{WR}$.

The ports are split into three functional groups:

- configuration group
- device control group
- ID registers.

Address lines A4 and A3 select the group, and A2 through A0 select the register in the group.

CONFIGURATION			DEVICE CONTROL 1		
Address	**Assignment**	**R/W**	**Address**	**Assignment**	**R/W**
00000	SRESET	W	01000	PORTX	R/W
00001	SISR	R/W	01001	PORTY	R/W
00010	SIMR	R/W	01010	PORTZ/AMC	R/W
00011	SEQSTS/PYC	R/W	01011	SEQCTL	R/W
00100	CSERR	R/W	01100	START	R/W
00101	CSCTL	R/W	01101	LOOP	R/W
00110	CSVAL	R/W	01110	ECCCTL	R/W
00111	CSCNT	R/W	01111	SECCNT	R/W
DEVICE CONTROL 2			**ID REGISTERS**		
Address	**Assignment**	**R/W**	**Address**	**Assignment**	**R/W**
10000	ECCP	W	11000	ID0	R/W
10001	ECCS	R	11001	ID1	R/W
10010	SPORT	R	11010	ID2	R/W
10011			11011	ID3	R/W
10100			11100	ID4	R/W
10101			11101	ID5	R/W
10110	TEST - do not use		11110	ID6	R/W
10111	SKIP	W	11111	ID7	R/W

TABLE 6-1. REGISTER GROUPS

6.2 REGISTER DEFINITIONS

The following sections describe each register in detail. The use of the register is described in general and/or each bit is described. The table below defines how bit directions are defined in these sections. Sometimes, a bit encoding is used to select a function that is not obvious from the definition of the bits involved. Refer to the section on Programming Notes for descriptions of these special modes.

The following table contains bit direction definitions.

Direction	Meaning
R	indicates that the bit is read only
W	indicates that the bit is write only
R/W	indicates that the bit may be written and read
R/C	indicates that the bit may be read and cleared by writing a one to that bit
C	indicates that the bit may be cleared by writing a one to that bit

6.3 CONFIGURATION GROUP

The configuration group is used to do initial set up of ports and clocks, handle interrupts, and set up the control store memory.

OSCDIV	CLKDIV	OSC	CPUCLK
0	0	XTAL/2	XTAL/6
0	1	XTAL/2	XTAL/4
1	0	XTAL	XTAL/3
1	1	XTAL	XTAL/2

6.3.1 SRESET - Set Hardware Reset Register (00)

Bits 7-1 are cleared to zero by reset. Bit 0 is set to one by an external reset. When writing one to SRST, any data on bits 7-1 are lost.

REGISTER	BIT	DIR	DEFINITION
00	7	W	CLKDIV: CPUCLK divisor (see Note 1 below)
00	6	W	OSCDIV: OSC divisor (see Note 1 below)
00	5	W	Unused.
00	4	W	Unused.
00	3	W	Unused.
00	2	W	REQTIM: buffer request timing. When this bit is one, REQA is set at the same time that the internal buffer data holding register is ready. When this bit is zero, REQA occurs one RRCLK period early. This aids in interfacing to certain buffer circuits that have a lag in their response times.
00	1	W	ID3$4: select ID address size (see Note 2)
00	0	W	SRST: hardware reset. This bit is set by an external reset on the $\overline{RST}$ input, or can be set by writing one to this bit. This bit must be set to zero before operating the WD10C01A, or before setting any of the other bits in this register or any other register.

TABLE 6-2. SRESET - SET HARDWARE REGISTER (00)

NOTE 1: The CPUCLK dividers are prescaled by the OSC dividers, as shown in the table below (XTAL is the clock generated by a crystal at X1 and X2):

NOTE 2: The eight ID register bytes are split into two fields: the first four are address, and auto-increment for each sector; and the second four are flag bytes which do not increment. The ID3$4 bit selects whether 3 or 4 bytes of ID address bytes will be used in the ID field. In 3 byte address fields, register ID0 is ignored. See Table 6-2A.

Two other considerations:

- CPUCLK = XTAL/3 will not be a 50% duty cycle unless XTAL is also a 50% duty cycle clock.
- When resetting the WD10C01A under CPU control, and the CPU is clocked by CPUCLK or OSC, set CPUDIV and OSCDIV back to zero before setting SRST. If this is not done, the clocks could glitch and cause the CPU to fail.

ID3$4	Address Counter Size	First Register In Field
0	4 bytes	ID0
1	3 bytes	ID1 (ID0 not used)

TABLE 6-2A. ID REGISTER SELECTION

6.3.2 SISR - Interrupt Status Register (01)

The interrupt status register is designed to be used in interrupt or polled mode. The status is not affected by the interrupt mask register (see below). The interrupting condition has precedence over the CPU clear, which is performed by writing a one to the selected interrupt bit. If the condition still exists, the clear will not be successful. The COMPLT and FAULT interrupts will stay set until the cause of the interrupt goes away. The IDFULL, DXFER, SECEND, and SM$IX interrupts are generated by single bit time pulses that are triggered by the leading edge of the interrupt cause, and can therefore be cleared immediately. SEQSTP is a direct status signal and is cleared when the sequencer is executing a command.

Before checking any bit (except SEQSTP), it should be cleared by writing a one to it. This register is not affected by reset.

REGISTER	BIT	DIR	DEFINITION
01	7	R	GINT - group interrupt. This is the state of the $\overline{INT}$ output, which is the logical OR of all of the enabled (by SIMR, see below) interrupt sources in this register.
01	6	R/C	IDFULL - ID registers full. This interrupt is set at the end of any ID field access by the transition of the control store ID bit from true to false. This interrupt should be serviced before the next ID field is accessed. This interrupt can be cleared immediately.
01	5	R/C	DXFER - data transfer started. This interrupt is set at the start of the data field by the transition of the control store BUFF or NOXFER bit from false to true. This interrupt can be used to determine when it is safe to write to the LOOP, SKIP, SECCNT, or ID registers, if necessary. This interrupt can be cleared immediately.
01	4	R/C	COMPLT - complete. This interrupt is set in response to the COMPLT input pin going true. This interrupt cannot be cleared until the COMPLT input pin goes false.
01	3	R	SEQSTP - sequencer stopped. This interrupt is set when the sequencer has stopped executing a command. This interrupt is cleared when the sequencer starts a new command.
01	2	R/C	SECEND - sector end interrupt. This interrupt is set by the leading edge of the LAST bit in the control store (see below). This is used to signal the end of a sector for buffer management and other overhead processing. The size of the field in which the LAST bit is set can be adjusted to match the processing overhead to the end of the sector for maximum CPU performance; the minimum size for this field is two bytes. This interrupt can be cleared immediately.
01	1	R/C	SM$IX - sector mark or index passed. This interrupt is set by the leading edge of the SECTOR or INDEX input going true, as selected by the mask in the sequencer control register (SEQCTL). RRCLK must be present for this interrupt to function. This interrupt can be cleared immediately.

TABLE 6-3. SISR - INTERRUPT STATUS REGISTER

REGISTER	BIT	DIR	DEFINITION
01	0	R/C	FAULT - drive fault. This interrupt is set in response to the DRVFLT input pin going true. This interrupt cannot be cleared until the DRVFLT input pin goes false.

TABLE 6-3. SISR - INTERRUPT STATUS REGISTER (CONTINUED)

6.3.3 SIMR - Interrupt Mask Register (02)

The interrupts listed are described in the SISR description. Writing a one to the mask bit enables the interrupt. The state of the mask bits does not affect the reading of status in SISR in any way. This register is cleared to zero by reset (interrupts disabled). Disabling GINT overrides any other enables set in this register.

REGISTER	BIT	DIR	DEFINITION
02	7	R/W	GINT - enable all interrupts
02	6	R/W	IDFULL - enable IDFULL interrupt
02	5	R/W	DXFER - enable DXFER interrupt
02	4	R/W	COMPLT - enable COMPLT interrupt
02	3	R/W	SEQSTP - enable SEQSTP interrupt
02	2	R/W	SECEND - enable SECEND interrupt
02	1	R/W	SM$IX - enable SM$IX interrupt
02	0	R/W	FAULT - enable FAULT interrupt

TABLE 6-4. SIMR - INTERRUPT MASK REGISTER

6.3.4 SEQSTS - Sequencer Status (03)

The following status bits are further clarified in later registers and sections. These bits can be used to check what sequencer operation is taking place.

REGISTER	BIT	DIR	DEFINITION
03	7	R	DATFLD - data field active. This bit means that the either the BUFF or the NOXFER bit is now active from the control store.
03	6	R	ECCEN - checksum calculation active. The WD10C01A is now calculating the checksum. This status line usually encompasses the ID or data field and checksum bytes, and is active during both read and write.
03	5	R	LAST - The LAST bit is now active from the control store.
03	4	R	ID - The ID bit is now active from the control store.
03	3	R	CHK - The CHK bit is now active from the control store. The WD10C01 is now processing the checkbytes or syndromes for the CRC or ECC.
03	2	R	WAIT - The wait sequencer is waiting for an event such as index, sector mark, address mark, or byte sync.
03	1	R	AMDET - Address mark detect. This is the raw AMDET pin input. This pin can be used as an extra input bit in designs (such as ST506) that do not require this function.
03	0	R	SEQOUT - The SEQOUT bit is now active from the control store.

TABLE 6-5. SEQSTS - SEQUENCER STATUS REGISTER

6.3.5 PYC - PORT Y Configuration (03)

This port is used to configure each port Y bit for input or output. This register is cleared to zero by reset, which causes all port Y pins to become inputs.

REGISTER	BIT	DIR	DEFINITION
03	3	W	PY3DIR - bit 3 direction: 0 = in; 1 = out
03	2	W	PY2DIR - bit 2 direction: 0 = in; 1 = out
03	1	W	PY1DIR - bit 1 direction: 0 = in; 1 = out
03	0	W	PY0DIR - bit 0 direction: 0 = in; 1 = out

TABLE 6-6. PYC - PORT Y CONFIGURATION (03)

6.3.6 Control Store Windows

The heart of the WD10C01A is its control store memory, which is organized as 32 instruction words of 28 bits each. Each 28 bit instruction is divided into three 8 bit parts: the control byte, the value byte, and the count byte; and one 4 bit part: the error control byte. The window registers are used to access these bytes. The control store address is specified by writing to the START register. Any write to a control store window register causes the START register to automatically increment to the next address to facilitate loading.

The contents of the control store are not affected by reset.

6.3.6.1 CSERR - Control Store Error Control Byte Window (04)

The transition of DAC from false to true, together with WG (in the control byte of the control store) or RCMP (read compare enable in the SEQCTL register), causes a one byte prefetch from the buffer. If an immediate fill character is used to specify the format data field or read compare byte (see below), and DAC is used to select the ECC for the data field, prefetch REQA signals are generated anyway.

REGISTER	BIT	DIR	DEFINITION
04	3	R/W	FAIL - Enable error failure. Setting this bit to one causes a command stop if an error is detected while executing the current control store instruction. The error can be CRC/ECC error, data miscompare error or a parity error during write operation.
04	2	R/W	RTY - Enable read error retry. This bit is valid only during read. Setting this bit to one causes a sector retry (see below) if a read error is detected while executing the current control store instruction. An example of the use of this bit is to cause a retry on an ID field miscompare.
04	1	R/W	DAC - Data field active. Set this bit to one when the current control store instruction involves a data field operation. This signal is used to select the data field checksum (RS-ECC) and to control data prefetch. When this bit is zero an ID field operation is assumed, and the ID field checksum (CRC) is selected.
04	0	R/W	SEQOUT - user defined output. This is tied to the SEQOUT output pin through some delays which align the signal to the byte boundary of the read or write serial data. This output can be used to control an external ECC circuit, such as a Reed-Solomon code circuit.

TABLE 6-7. CSERR-CONTROL STORE ERROR CONTROL BYTE WINDOW

6.3.6.2 CSCTL - Control Store Control Byte Window (05)

REGISTER	BIT	DIR	DEFINITION
05	7	R/W	SVSEL - value byte select. This bit selects the use of the value byte, which can be either immediate data (SVSEL = 0) or a select for data source or destination (SVSEL = 1).
05	6	R/W	CWSEL - count byte select. The bit selects the use of the count byte, which can be either an immediate byte count (CWSEL = 0) or to specify an external condition to wait for (CWSEL = 1).
05	5	R/W	WG - write gate output. This is tied directly to the WG output pin.
05	4	R/W	RG - read gate output. This is tied to the retry logic, which is then tied to the RG output pin.
05	3	R/W	AM - address mark enable output. This is tied to the AMC register (see below) which generates the AMENA signal.
05	2	R/W	CMPEN - compare enable. This signal is used to indicate that the bytes of the currently selected data source are to be compared with incoming serial data (ID, marker bytes, buffer data).
05	1	R/W	SKPEN - jump to SKIP address at end of the current instruction. This causes an absolute jump to the SKIP register address when the current control store instruction is finished. Typically, this is used to set up a read and write program in the control store with a common ID search routine. (See later examples and SKIP register definition.)
05	0	R/W	JMPEN - jump to LOOP address at end of the current instruction. This causes a conditional jump to the LOOP register if the sector count (SECCNT) is not zero. If SECCNT is zero, the next sequential instruction is executed. Typically, this is used to specify the end of a sector, and tells the sequencer to go to the LOOP register address to operate on the next sector.

TABLE 6-8. CSCTL - CONTROL STORE BYTE WINDOW (05)

6.3.6.3 CSVAL - Control Store Value Byte Window (06)

The use of the value byte depends on the state of the SVSEL bit of the control byte. When SVSEL is zero, the value byte specifies actual immediate data, like address mark, gap, and PLL sync bytes. When SVSEL is one, the value byte becomes an encoded bit field that enables the correct data source or destination. Both uses are shown below.

To get large sector sizes, multiple control store instructions are used. For example, for a 1024 byte sector size, use four instructions with 256 byte count fields. With this scheme, the last instruction must be flagged for error correction and write prefetch purposes. The LAST bit must be set with the BUFF or NOXFER bit to ensure proper operation.

REGISTER	BIT	DIR	DEFINITION
06	7-0	R/W	VALUE7-0 (actual value for field)

TABLE 6-9. SVSEL of CONTROL BYTE ZERO

REGISTER	BIT	DIR	DEFINITION
06	7	R/W	BUFF - data buffer. This bit causes data to be transferred to (disk read) or from (disk write or read compare) the disk data buffer. The leading edge of BUFF causes a decrement of the SECCNT register, and increments the ID write register counters (see below).
06	6	R/W	NOXFER - no data transfer. This bit is used when an ECC/CRC verify on the data field with no buffer data transfer is being performed. This bit is set INSTEAD of the BUFF bit, and affects the SECCNT and ID registers in the same way as the BUFF bit.
06	5	R/W	LAST - last data buffer xfer control store instruction. Long data fields (greater than 256 bytes) are specified by using multiple control store instructions (2 for 512 bytes, 4 for 1024 bytes, etc.). When the control store instruction is the last instruction of the data field specifiers, the LAST bit must be set to flag this. This only applies to data buffer transfers. LAST is set in ADDITION to BUFF or NOXFER. The CSCNT byte must be set to at least 01 when this bit is used.
06	4	R/W	R/W ID - ID registers. On ID read, the incoming ID field from the disk is compared against the ID write registers and written at the same time to the ID read registers. On write (format), the data source is the ID write registers.
06	3	R/W	CHK - checksum field. On read, this starts the check for a correct checksum. On write, this causes the checksum shift register to be gated into the NRZ0 data. In either case, the calculation is halted at the end of this instruction. The DAC bit in CSERR window selects the appropriate checksum automatically.

TABLE 6-10. SVEL of CONTROL BYTE ONE

6.3.6.4 CSCNT - Control Store Count Byte Window (07)

The use of the count byte depends on the state of the CWSEL bit of the control byte. When CWSEL is zero, the count byte specifies the actual length of that field in bytes. When CWSEL is one, the count byte becomes an encoded bit field that specifies a condition to wait for before proceding. Both uses are shown below.

The count value is set to the actual number of bytes to do minus one. Therefore, 00 denotes a one byte field, and FF a 256 byte field.

REGISTER	BIT	DIR	DEFINITION
07	7-0	R/W	COUNT7-0 (actual size of field - 1)

TABLE 6-11. CWSEL of CONTROL BYTE ZERO

REGISTER	BIT	DIR	DEFINITION
07	7	R/W	WDAM - Wait for data address mark (ST412 ONLY). This bit causes the WD10C01A to pause until a data address mark is detected at the AMDET input. Bits 6-0 specify a timeout count that is the maximum number of byte times from the end of the ID field to the data address mark. The value field is used to specify the data pattern to compare against for byte sync.
07	6	R/W	WIAM - Wait for ID address mark (ST412 ONLY). This bit causes the WD10C01A to pause until an ID address mark is detected at the AMDET input. The value field is used to specify the data pattern to compare against for byte sync.
07	5	R/W	WIX - Wait for index. Pauses until index is detected at the INDEX input.
07	4	R/W	WSM - Wait for sector mark. Pauses until sector mark is detected at the SECTOR input.
07	0	R/W	STOP - Stop immediate. This bit causes the command sequencer to immediately turn off all control outputs and return to the stopped state.

TABLE 6-12. CWSEL of CONTROL BYTE ONE

6.3.6.5 Wait Condition Sequences

The following sequences are performed on the above wait conditions to ensure proper error handling:

WDAM:

Wait for AMDET. If the byte count is exceeded, a sync error is recorded, and (if the control store FAIL bit is set) the command halts. When AMDET is detected correctly, the sequencer waits sixteen bit times for the sync byte in the value byte to be matched. The bit counter is decremented on each bit while the AMDET signal is active. If this bit count is exceeded, this is also a sync error. When a sync error occurs, the sequencer will stop or retry as defined by the control store FAIL and RTY bits. Checksum calculation starts when the sync byte is detected, and the sync byte is included in the calculation.

WIAM:

Wait for AMDET. Since ID address marks could be found anywhere, there is no byte count, and the CPU must perform its own timeout. However, after AMDET is detected, the sequencer waits sixteen bit times for the sync byte in the value byte to be matched. The bit counter is decremented on each bit while the AMDET signal is active. If this bit count is exceeded, this is considered a sync error, and (if the control store RTY bit is set) a retry is performed (ID retry is discussed in a later section). Checksum calculation starts when the sync byte is detected, and the sync byte is included in the calculation.

WIX and WSM:

The signal (index or sector mark) is waited on forever. The CPU must perform its own timeout and issue the KILL bit to the SEQCTL register to stop the command (see below). To do a wait for either index or sector mark (typical on hard sector drives), set both WIX and WSM.

6.4 DEVICE CONTROL GROUP

The device control group is used to set and read port X, Y, and Z bits for drive control. It is also used to execute data transfer commands with the sequencer, and correct ECC errors.

6.4.1 PORTX - PORTX Output Bits (08)

This register is cleared to zero by reset. The state of the output bits may be read back, allowing for read/modify/write operation.

REGISTER	BIT	DIR	DEFINITION
08	7-0	R/W	PX7-0 (direct to output pins)

TABLE 6-13. PORTX OUTPUT BITS (08)

6.4.2 PORTY - PORT Y I/O Bits (09)

The direction of each bit is controlled by the Port Y Control Register (03). If a particular bit is set for input, then the state of the external pin is read by a read of this port. If the bit is set for output, then the last state written to that bit is read (same as Port X). Writing to this register does not affect bits configured for input. Reset sets all bits to input, but does not affect the output latched data.

REGISTER	BIT	DIR	DEFINITION
09	3-0	R/W	PY3-0 (direct to I/O pins)

TABLE 6-14. PORTY I/O BITS (09)

6.4.3 PORTZ - PORT Z INPUT BITS (0A)

REGISTER	BIT	DIR	DEFINITION
0A	5-0	R	PZ5-0 (direct from input pins)

TABLE 6-15. PORTZ INPUT BITS (0A)

6.4.4 AMC - Address Mark Control (0A)

The AMC register defines during which bit times the AMENA signal is active. When AM is set in the control store control byte of the current instruction, the AMENA signal will be active during the bit times specified in this register. This register is not affected by reset. Note that the bits in this register are ordered backwards from the data bus numbering; i.e., writing 01 to this register will turn on AMENA when bit 7 of the serial data stream is active on NRZO.

REGISTER	BIT	DIR	DEFINITION
0A	7-0	W	Address mark bit control bit 0-7

TABLE 6-16. AMC - ADDRESS MARK CONTROL (0A)

6.4.5 SEQCTL - Sequencer Control Register (0B)

REGISTER	BIT	DIR	DEFINITION
0B	7	W	RGERLY - read gate early bit. This bit controls the timing of RG signal. If the bit is set to zero, the RG signal is extended automatically until the ECC/CRC calculation is finished. Setting this bit to one will disable this feature.
0B	3	W	IXMASK - index interrupt mask. This bit is used with the SMMASK bit to select the conditions for the SM$IX interrupt (see table below).
0B	2	W	SMMASK - sector mark interrupt mask (see table below).
0B	1	W	RCMP - enable read compare. This bit MUST be set when performing a read data compare command with the data buffer. It is used to change the buffer access from write buffer to read buffer.
0B	0	W	KILL - immediately kill the currently executing command. The sequencer will stop within three byte times. This bit must be asserted for at least 2 bytes time.
0B	5	R	ECCERR - checksum error. An ECC or CRC error has been detected. This bit is valid when the sequencer is stopped. If the control bit IGNERR in ECCCTL is turned on, this status bit will always be zero and the sequencer will continue its operation ignoring the error condition.
0B	4	R	IDERR - ID checksum error. A CRC ID field error occurred during the previous ID field read. This condition is latched when a checksum error causes an ID retry, and is cleared at the start of the next ID field read (ID false to true transition).
0B	3	R	PTYERR - parity error. A parity error during a transfer from the data buffer to the WD10C01A occurred during the previous command. If FAIL bit is set in the current CS instruction, the sequencer will stop its operation at the end of current instruction and WG output will be deasserted. This bit is valid only after the sequencer is stopped.
0B	2	R	SYNCER - sync search error. The search for a data field marker byte or sync byte failed. This bit is valid when the sequencer is stopped.
0B	1	R	CMPERR - compare data error. A field that was compared against some specified data source compared incorrectly. This bit is valid when the sequencer is stopped.

TABLE 6-17. SEQCTL - SEQUENCER CONTROL REGISTER (0B)

This register is not affected by reset. The IXMASK and SMMASK select the conditions for the SM$IX interrupt. Note that these mask bits are used to select only the source, and do not affect or enable the external interrupt signal. The table below defines the use of the SM$IX source mask bits:

SMMASK	IXMASK	SM$IX set true on leading edge of:
0	0	nothing
0	1	INDEX
1	0	SECTOR
1	1	INDEX OR SECTOR

6.4.6 START - Sequencer Start Address (0C)

The START register is not affected by reset. This register is actually two devices: the START holding register, and the control store address counter. When the CPU writes to this port, the data bus value is latched into both the holding register and the address counter. The holding register is affected only by CPU writes, but the counter is incremented or reloaded by several different events. The START register is used for the following functions:

1. The START register specifies the starting address in the control store for the program loaded there. When the CPU writes to the SECCNT register (see below), and the sequencer is stopped, the sequencer loads the contents of the holding register into the address counter and starts with the instruction at that location.

2. Reading this port gives the current contents of the control store address counter. Reading the counter while the sequencer is running indicates which control store instruction is currently being executed. It is recommended that the CPU 'debounce' this port by reading the START register until the same value is read twice. This prevents erroneous values being read at transition times. Reading the counter when the sequencer is stopped indicates where in the control store the condition causing the last halt occurred. This can be used to determine in which instruction an ECC or other fatal error occurred.

3. The START register is used to specify the current address for control store window access. See control store windows above. The control store address counter is automatically incremented after any write to a control store window register (CSERR, CSCTL, CSVAL, or CSCNT).

4. Writing to the START register clears error conditions. When a read error occurs, causing the command to halt (see FAIL bit in CSERR above), the error status is latched in the SEQCTL register. A write to the START register clears ECCERR, I0E-I4E, CERR, EERR, SYNCER, CMPERR, and also PTYERR (which does not cause a halt). This must be performed prior to issuing any new commands. If SECCNT is non-zero, the sequencer will start as soon as the error is cleared.

While the sequencer is running, the control store address counter is the program counter, and points at the current instruction being executed. The address in the counter can be changed by the following events during program execution:

1. At the end of an instruction that has the SKPEN bit set, the address counter is loaded with the contents of the SKIP register.

2. At the end of an instruction that has the JMPEN bit set, if the SECCNT register is not zero (i.e., more sectors to do), the address counter is loaded with the contents of the LOOP register. If SECCNT is zero, the address counter is incremented and the next sequential instruction is executed.

REGISTER	BIT	DIR	DEFINITION
0C	4-0	R/W	START4-0

TABLE 6-18. START - SEQUENCER START ADDRESS (0C)

3. If an instruction has the RTY bit set, and an error flag is currently set (ECCERR, CMPERR, or SYNCER), a retry occurs and the address counter is loaded with the contents of the LOOP register.

4. If a program halts due the STOP bit being set (i.e., the normal end of the program with SECCNT zero), and the CPU writes to the SECCNT register with more blocks to do before it detects the stopped condition, the sequencer will restart using the current value of the START holding register. This prevents erroneous restarts from occurring.

5. In all other cases, the control store address counter is incremented and the next sequential instruction is executed.

6.4.7 LOOP - Sequencer Loop Address (0D)

The LOOP register specifies the address to set the START value to at the end of a control store instruction that has the JMPEN bit set in the control byte. This also happens when an ID retry occurs (see below). The use of the LOOP register is based on the idea that all sector operations are sequential in nature, and that when the operation is complete, a single jump back to the start of the sequential sector operation is all that is needed. This register is not affected by reset.

REGISTER	BIT	DIR	DEFINITION
0D	4-0	W	LOOP4-0 (address to loop to in loop)

TABLE 6-19. LOOP - SEQUENCER LOOP ADDRESS (0D)

6.4.8 ECCCTL - Error Correction Control Register (0E)

CRCSET and ECCCLR must be set high and then low to complete the preset or clear operation. All bits in this register are cleared to zero upon reset time.

REGISTER	BIT	DIR	DEFINITION
0E	5	W	ECCCLR - clear CRC and ECC shift register. Writing one to this register causes the CRC and ECC shift register to be held in the clear (all bits zero) state. The CPU must then write zero to this bit to complete the clear pulse.
0E	4	W	CRCSET - preset CRC shift register. Writing one to this register causes the CRC shift register to be held in the preset (all bits one) state. The CPU must then write zero to this bit to complete the preset pulse.
0E	3	W	CRCNIT - CRC shift register initial state. This bit selects whether to start the CRC shift register with all zeros (CRCNIT = 0) or all ones (CRCNIT = 1) when starting checksum calculation.
0E	2	W	IGNERR - ignore CRC/ECC error. Forces ECCERR bit in the SEQCTL register to zero, and causes the read error logic to ignore this error by continuing with its operation. The status bit EERR/CERR in the ECCS register is not affected by this control bit and will be set on error.
0E	1	W	DISPTY: disable parity checking. Setting this bit to one forces the PTYERR bit in the SEQCTL register to zero. Even if the FAIL bit in the current CS instruction is set, the sequencer will continue its operation normally on parity error.

TABLE 6-20. ECCCTL - ERROR CORRECTION REGISTER (0E)

6.4.9 SECCNT - Sector Count Register (0F)

The sector count register is used to specify the number of iterations to perform the operation programmed into the control store. Writing any non-zero value to the SECCNT register causes the sequencer to start, so therefore the range of possible sector counts is from 1 to 255. Writing a zero to SECCNT during the data field will cause the sequencer to halt at the end of the current sector. The sector count is automatically decremented at the start of the data field, after the byte sync or address mark character(s) have been detected (leading edge of BUFF or NOXFER). The SECCNT register can be written to at any time except the start of the data field.

This register is cleared to zero by reset, or when the KILL bit in the SEQCTL register is set. It is NOT cleared by a halt on error, so that the sector in error can be determined.

REGISTER	BIT	DIR	DEFINITION
0F	7-0	R/W	Sector count value

TABLE 6-21. SECCNT - SECTOR COUNT REGISTER (0F)

6.4.10 ECCP- ECC Parameter Register (10)

REGISTER	BIT	DIR	DEFINITION
10	3	W	SYNCECC - include sync byte in the ECC calculation. This bit is initialized to one upon reset. If this control bit is set to one, the ECC calculation begins with the first data sync byte. Otherwise, the first data sync byte is excluded from the ECC calculation.
10	2	W	WSYNCCRC - include sync byte in the CRC calculation. This bit is initialized to one upon reset. If this control bit is set to one, the CRC calculation begins with the first ID sync byte. Otherwise, the first data sync byte is excluded from the CRC calculation.
10	1	W	DEG6/5 - RS ECC degree control bit. This bit is initialized to zero upon reset. If this bit is set to zero, degree 5 polynomial is selected, otherwise degree 6 will be used.
10	0	W	WIFS5/3 - Interleave select. This bit is initialized to zero upon reset. Setting this bit to one will select five way interleave format, otherwise three way format will be used.

TABLE 6-22. ECCP - ECC PARAMETER REGISTER (10)

6.4.11 ECCS - RS-ECC Status Register (11)

REGISTER	BIT	DIR	DEFINITION
11	6	R	EERR - ECC error detected. This bit is set whenever an ECC error is detected in any one of the interleaves. At least one of the status bits I4E - I0E will also be set. This bit is cleared to zero after reset and before another operation is started. This bit is not affected by the control bit IGNERR, which prevent the status ECCERR from being set.
11	5	R	CERR - CRC error detected. This status bit is set whenever a CRC error is detected in the ID field. This bit is cleared upon reset. The control bit IGNERR does not affect this status bit.
11	4	R/W	I4E - ECC error in interleave 4. This bit is set only when an ECC error is detected in the interleave 4. The microprocessor can read the syndrome bytes from this interleave only when this status bit is set. This status bit is cleared upon reset. It also be cleared by writing a zero into this bit.
11	3	R/W	I3E - ECC error in interleave 3. This bit is set only when an ECC error is detected in the interleave 3. The microprocessor can read the syndrome bytes from this interleave only when this status bit is set. This status bit is cleared upon reset. It also be cleared by writing a zero into this bit.
11	2	R/W	I2E - ECC error in interleave 2. This bit is set only when an ECC error is detected in the interleave 2. The microprocessor can read the syndrome bytes from this interleave only when this status bit is set. This status bit is cleared upon reset. It also be cleared by writing a zero into this bit.
11	1	R/W	I1E - ECC error in interleave 1. This bit is set only when an ECC error is detected in the interleave 1. The microprocessor can read the syndrome bytes from this interleave only when this status bit is set. This status bit is cleared upon reset. It also be cleared by writing a zero into this bit.
11	0	R/W	I0E - ECC error in interleave 0. This bit is set only when an ECC error is detected in the interleave 0. The microprocessor can read the syndrome bytes from this interleave only when this status bit is set. This status bit is cleared upon reset. It also be cleared by writing a zero into this bit.

TABLE 6-23. ECCS - RS-ECC STATUS REGISTER (11)

6.4.12 SPORT - Syndrome Port (12)

The microprocessor reads the syndrome bytes for each interleave through this port. The WD10C01A automatically transfers the syndromes of the next interleave in error after all the syndromes in the current interleave are read. The microprocessor must read (interleave number * ECC degree) times from this port to complete a syndrome transfer from an interleave. The syndrome bytes are transferred starting with the **highest** byte from the **lowest** interleave number. Only the syndromes from an interleave in error are transferred. If the microprocessor clears the error flag from an interleave, the syndromes from that interleave cannot be read, but the content of the syndrome registers are not destroyed. Those can be read by reseting the error flag to one.

The syndrome bytes are valid only when sequencer is stopped. The syndrome registers are cleared upon reset.

6.4.13 TEST - Test Register (16)

This register is intended for test purpose in the manufacturing and must not be used in normal operations. It is described here only for reference.

There are three test functions implemented in this register. First, writing into this register with any data, will set the OSC and CPUCLK outputs in a predetermined state. During the write cycle, the OSC output is set to 0 level and the CPUCLK output is set to a 1 level.

Second, the microprocessor can write 080H into this register to disable the output signal WG. This signal will remain inactive until the microprocessor writes a 00H into the register.

Third, the microprocessor can write 0C0H into the register to increment the ID registers. Subsequently, the microprocessor must write 00H, before it can continue to increment the registers again. The output WG is also disabled during the test operation.

REGISTER	BIT	DIR	DEFINITION
12	7-0	R	SYNDR7-0 - Syndrome bit7-bit0.

TABLE 6-24. SPORT - SYNDROME PORT (12)

6.4.14 SKIP - Skip Address Register (17)

This register is not affected by reset. The SKIP register is used to perform an absolute jump to the location specified at the completion of a control store instruction that has the SKPEN bit set to one. This can typically be used to define whether the current operation is to be a sector read or a sector update write. Consider the following generalized control store program that has three parts: ID search, read data field, and write data field.

When the SKIP register is set to address 05, the WD10C01A will perform a sector read operation. When the SKIP register is set to address 0C, the WD10C01A will perform a sector write operation. A more detailed example can be found in a subsequent section of this document.

REGISTER	BIT	DIR	DEFINITION
17	4-0	W	SKIP4-0

TABLE 6-25. SKIP - SKIP ADDRESS REGISTER (17)

ADDRESS	CONTROL STORE OPERATION
00	ID SEARCH
01	ID SEARCH
02	ID SEARCH
03	ID SEARCH
04	ID SEARCH (SKPEN bit set)
05	READ DATA FIELD
06	READ DATA FIELD
07	READ DATA FIELD
08	READ DATA FIELD
09	READ DATA FIELD
0A	READ DATA FIELD
0B	STOP
0C	WRITE DATA FIELD
0D	WRITE DATA FIELD
0E	WRITE DATA FIELD
0F	WRITE DATA FIELD
10	WRITE DATA FIELD
11	WRITE DATA FIELD
12	STOP

TABLE 6-26. SKIP CONTROL STORE EXAMPLE

6.5 ID REGISTER GROUP

The eight ID registers (ID0 - ID7) are used to set the ID field during format and read/write operations. Writing to the ID registers sets the field to search for, or to write during format. These registers are refered to as the ID write registers. The first four ID write registers are set up as a 32-bit counter, and they automatically increment at the start of the data field. The second four are simple registers and are intended for use as flag and defect indicators.

The ID read registers contain the last ID field read from the media (valid when IDFULL in SISR is true). These registers must be read when the sequencer is NOT reading an ID field, or else the CPU will not read the correct value.

REGISTER	BIT	DIR	DEFINITION
18	7-0	R/W	ID0 bits 7-0 (counter MSbyte)

TABLE 6-27. ID0 - ID REGISTER 0 (18)

REGISTER	BIT	DIR	DEFINITION
19	7-0	R/W	ID1 bits 7-0 (counter)

TABLE 6-28. ID1 - ID REGISTER 1 (19)

REGISTER	BIT	DIR	DEFINITION
1A	7-0	R/W	ID2 bits 7-0 (counter)

TABLE 6-29. ID2 - ID REGISTER 2 (1A)

REGISTER	BIT	DIR	DEFINITION
1B	7-0	R/W	ID3 bits 7-0 (counter LSbyte)

TABLE 6-30. ID3 - ID REGISTER 3 (1B)

REGISTER	BIT	DIR	DEFINITION
1C	7-0	R/W	ID4 bits 7-0 (flag)

TABLE 6-31. ID4 - ID REGISTER 4 (1C)

REGISTER	BIT	DIR	DEFINITION
1D	7-0	R/W	ID5 bits 7-0 (flag)

TABLE 6-32. ID5 - ID REGISTER 5 (1D)

REGISTER	BIT	DIR	DEFINITION
1E	7-0	R/W	ID6 bits 7-0 (flag)

TABLE 6-33. ID6 - ID REGISTER 6 (1E)

REGISTER	BIT	DIR	DEFINITION
1E	7-0	R/W	ID7 bits 7-0 (flag)

TABLE 6-34. ID7 - ID REGISTER 7 (1F)

6.6 DEVICE PROGRAMMING

6.6.1 Initialization

1. Write one and then zero to the SRST bit of the SRESET register to complete the device reset sequence.

2. Write the remainder of the bits of the SRESET register to the CPU and OSC clock speeds, and ID counter size and data request timing.

3. Clear all interrupts that can be cleared by writing 0FFH to the SISR (some may still be set because they are level true). Write the initial mask to the SIMR.

4. Configure port Y.

5. Configure the address mark control bits using the AMC register.

6. Set up the ECC by programming the ECCCTL and ECCP registers. Set CRCNIT in the ECCCTL register to the appropriate value for a zero seed or a one's seed.

7. Load the control store program into the control store.

6.6.2 Command Programming

1. Load the control store with the particular instructions appropriate for the command desired (format, read, write, etc.), if necessary.

2. Set the START, LOOP, and SKIP registers to the appropriate values for the control store program.

3. Write the number of sectors to do to SECCNT. This also starts the command sequencer.

4. Clear the interrupt status bits, and set the mask.

5. Wait for SEQSTP. Check the SEQCTL and ECCS status bits for an error.

6.6.3 Control Store Programming

There are a few 'tricks' in programming the WD10C01A control store that are not indicated by the discussions of the individual control bits:

READ BYTE SYNC

On formats that use a simple byte sync byte (like ESDI) for ID and data field markers, the search is performed by setting the control byte bits RG and CMPEN, and the count field is set with a maximum byte timeout count. The condition of no byte sync yet found, and RG * CMPEN set, defines this mode. AMDET is not used and has no effect on this operation. Checksum calculation starts when the sync byte is detected, and the sync byte is included in the calculation.

WRITE BYTE SYNC

Write byte sync is used to mark the start of the byte string for which the ECC/CRC checksum is calculated. To do this, set WG and CMPEN in the control byte. The start of the control store instruction with WG and CMPEN set clears the checksum register to the state defined by CRCNIT. The transition to the next instruction begins the checksum calculation. For example, in a ST412 drive the ID field might be defined by the byte string: A1 FE ID ID ID FLAG ECC ECC ECC ECC; with PLL sync before, and gap after. The control store instructions would be coded as follows.

ADDRESS	DATA	COUNT	CONTROL	ERROR
00	00	0B	WG,CMPEN	PLL sync field
01	A1	00	WG,AM	address mark
02	FE	00	WG	address marker
03	ID	02	WG,SVSEL	ID field
04	ID	00	WG,SVSEL	flag byte
05	CHK	03	WG,SVSEL	checksum
06	00	03	WG	pad and splice

TABLE 6-35. WRITE BYTE SYNC EXAMPLE

In the example, control store instruction 00 contains the WG-CMPEN combination that says to begin checksum calculation on instruction 01; therefore, the bytes defined in instructions 01 through 05 are included in the checksum calculation. The calculation is terminated by the CHK value code in instruction 05.

DATA FIELD SIZES

The data field size is set by using multiple control store instructions to build up the size in 256 (or less) increments. Two issues come up with this method:

1) The last data field instruction must be marked to ensure correct buffer data transfer timing and checksum calculation. The LAST bit is used with the BUFF or NOXFER bit to perform this marking.

2) For programming purposes, it is desirable to not have to change the control store significantly when changing sector size. To do this, allocate enough control store instructions to be able to set the largest sector size to be supported. Then, adjust the count fields in all of the instructions to get the desired sector size. For example, if a controller supports 256, 512, and 1024 byte sectors, use the programming shown in the example below:

Rel Addr	Counts for		
	256	512	1024
00	3F	7F	FF
01	3F	7F	FF
02	3F	7F	FF
03	3F	7F	FF

TABLE 6-36. DATA FIELD SIZE EXAMPLE

If the LAST bit is being used as an interrupt, the last instruction can be split into two parts. The second part contains the LAST bit, and can be adjusted to the required time before the end of the sector, since the leading edge of LAST generates the interrupt.

ID SEARCH AND FLAG BYTES

When specifying an ID field, it is desirable to have address bytes that are compared on a search, and flag bytes that are not. To do this, split the ID field in the control store into two separate instructions for address and flag. When performing an ID search, set the CMPEN bit only in the ID address instruction, and not in the ID flag instruction.

DATA CHECKSUM VERIFY

To do a data field ECC or CRC verify only command, set the data field source as NOXFER. No data will be transferred to the buffer.

DATA COMPARE VERIFY

To do a data field compare verify, set the RCMP bit in the SEQCTL register, and set the CMPEN bit in the data field control store instruction. Data will be transferred from the data buffer and compared with the incoming disk data.

FILL GAP TO END OF TRACK

To generate fill data through to the end of the track during format, set the WIX bit in the last control store instruction, and set WG and the fill character. The WD10C01A will fill in gap bytes until index occurs. This also works for gaps to sector marks.

6.6.4 Programming Examples

This section gives programming examples for ST412 format, read, and write commands. The following design parameters are assumed:

1. The ID field is three bytes of address, one byte of flag, and two bytes of CRC.
2. Intersector gaps are 20 bytes of 04EH data, PLL sync fields are 12 bytes of 000H.
3. ID address mark and marker byte are 0A1FEH, data address mark and marker byte are 0A1F8H.
4. Data field is protected through degree six ECC with interleave factor of five. There are 30 bytes of checksum.

The WD10C01A registers are programmed as follows:

SRESET: ID3$4 = 1 (3 bytes of address counter)

6.6.4.1 Format Track Example

ADDRESS	VALUE	COUNT	CONTROL	NOTES
00	00	WIX	CWSEL	start on index
01	4E	13	WG	post index gap
02	00	0B	WG, CMPEN	PLL sync
03	A1	00	WG,AM	address mark
04	FE	00	WG	address mark
05	ID	03	WG,SVSEL	ID address & flag
06	CHK	01	WG,SVSEL	CRC
07	00	02	WG	write splice
08	00	0B	WG, CMPEN	data PLL sync
09	A1	00	WG, AM, DAC	address mark
0A	F8	00	WG, DAC	address mark
0B	BUFF	7F	WG, SVSEL,DAC	data field
0C	BUFF	7F	WG, SVSEL,DAC	data field
0D	BUFF	7F	WG, SVSEL,DAC	data field
0E	BUFF, LAST	7F	WG, SVSEL,DAC	data field
0F	CHK	1D	WG, SVSEL,DAC	ECC
10	00	02	WG,JMPEN,DAC	write splice
11	4E	WIX	WG, CWSEL	pre-index gap
12	00	STOP	CWSEL	stop at end of track
START = 00, LOOP = 01, SKIP = NOT USED				

TABLE 6-37. FORMAT TRACK EXAMPLE CONTROL STORE

Line by line discussion of Format Track Example:

00: This instruction just waits until the leading edge of index. No writing is occurring. If there is no index signal working on the drive, the CPU will have to timeout and issue an abort. The DAC bit is zero to select the checksum for the ID, as specified by the SRESET register.

01: This instruction is the standard 4E gap from index to the start of the first sector. This instruction is also the intersector gap, because after the data field write splice on instruction 11 is written, the WD10C01A will jump to this address specified by the LOOP register.

02: This instruction is the PLL sync field for the ID field. The WG - CMPEN combination also marks the start of the CRC calculation starting with instruction 03.

03: This is the first address mark byte for the ID field. The AM bit generates AMENA with the AMC register, suppressing the clock bit in the A1 byte.

04: This is the second address mark byte, and is treated as simple immediate data for format purposes.

05: This is the ID address and flag field.

06: ID CRC field. The count field is set to (2-1) bytes for the ID CRC. The end of this instruction also stops the checksum calculation.

07: ID write splice and data PLL sync field. This gap protects the end of the ID field from the start of the data field on later update write commands.

08: Data PLL sync field. The WG - CMPEN combination also marks the start of the ECC calculation starting with instruction 0A. The DAC bit is set here to select the data field checksum, and in this case it also causes a one byte prefetch from the buffer.

09: This is the first address mark byte for the data field. The AM bit generates AMENA with the AMC register, suppressing the clock bit in the A1 byte.

0A: This is the second address mark byte, and is treated as simple immediate data for format purposes.

0B-0D: These are the first three instructions that define the sector data field. The data field is split into four control instructions so that switching between sector sizes will not require reordering the control store data (see above). The example shows a 512 byte sector.

0E: The last instruction of the data field marks the end of the data field using the LAST bit set with BUFF.

0F: Data ECC field. ECC degree six with interleave factor of five is used, the count field is set to (30-1) bytes. The end of this instruction also stops the checksum calculation.

10: Data field write splice. This instruction pads the end of the data field, making sure that the checksum is not lost by the drive. The JMPEN bit is also set in this instruction, meaning that the next instruction to be accessed will be the one addressed by the LOOP register, in this case 01. This happens only if SECCNT is not zero. If SECCNT is zero, the next control instruction is 11.

11: After the last sector, SECCNT is zero. After instruction 10 is done, the next instruction will be this one instead of instruction 01 (LOOP register). This instruction writes the 4E gap until index.

12: The STOP bit causes the sequencer to shut down immediately and turn off all external signals (like WG, etc.).

The CPU performs the FORMAT TRACK command as follows:

1. After setting up the control store and address registers as defined above, write the total sectors per track to the SECCNT register.

2. Write DXFER, FAULT, and SEQSTP true to the SISR register to clear those status bits, and to the SIMR register to enable them.

3. Wait until GINT is true with a timeout in case INDEX is not present on the interface. Check FAULT for error, and abort if true. If DXFER goes true, update the ID write registers with the next ID field (when using non-consecutive, i.e., not 1:1 interleave).

4. Repeat step 3 for all sectors on the track.

5. After the last sector, wait for SEQSTP to be true, indicating the end of the command.

6.6.4.2 Read Sector Example

ADDRESS	VALUE	COUNT	CONTROL	NOTES
00	A1	WIAM	RG,CWSEL, RTY	address mark search
01	FE	00	RG,CWSEL, RTY	ID marker byte
02	ID	02	RG, SVSEL, CMPEN	ID address
03	ID	00	RG, SVSEL	ID flag
04	CHK	01	RG, SVSEL	CRC
05	00	03	RTY	allow ID retry
06	A1	WDAM (1F)	RG, CWSEL, DAC, FAIL	address mark search
07	F8	00	RG, CWSEL, DAC, FAIL	data marker byte
08	BUFF	3F	RG, SVSEL, DAC	data field
09	BUFF	3F	RG, SVSEL, DAC	data field
0A	BUFF	3F	RG, SVSEL, DAC	data field
0B	BUFF, LAST	3F	RG, SVSEL,DAC	data field
0C	CHK	ID	RG, SVSEL,DAC	ECC
0D	00	02	JMPEN,DAC, FAIL	end of sector
0E	00	STOP	WG, SVSEL,DAC	stop at end of read
START = 00, LOOP = 00, SKIP = NOT USED				

TABLE 6-38. READ SECTOR EXAMPLE CONTROL STORE

Line by line discussion of Read Sector example:

00: This is the first instruction of the read sector command. The WIAM bit tells the WD10C01A to search for an ID address mark byte. The WD10C01A will not timeout, so the CPU will have to abort after some period of time, if necessary. The DAC bit is zero during the ID field to select the ID checksum. The RTY bit is set so that if a read error (in this case, no sync match after AMDET true) occurs, a retry will be performed.

01: This tells the WD10C01A to verify the ID marker byte. If it does not compare exactly, an ID retry occurs (see below) because the RTY bit is set.

02: This is the ID address field. The CMPEN bit causes a compare of the ID write registers with the incoming ID field. The incoming ID is also written to the ID read registers.

03: This is the ID flag field. This control store instruction is separate from the ID address field because the flag byte is not compared with the ID write register. It is still written to the ID read register.

04: ID CRC field. The count field is set to (2-1) bytes for the ID CRC. This instruction switches the checksum shift register into check mode.

05: This instruction is both a pad over the write splice and a check for an ID retry. The ID compare status and checksum error status are latched

and held, and the RTY bit is now set, allowing the checking of this status. An ID retry occurs if the ID did not match, or if there was a checksum error. The checksum error is latched into the IDERR bit for future reference. We waited until this instruction to check for an error so that the ID read registers could be loaded with this ID field.

06: The WDAM bit tells the WD10C01A to look for the data address mark. The (1F) indicates the byte count that is loaded into the byte count. This value is the maximum number of byte times that the WD10C01A will look for the address mark. This prevents locking up on a subsequent data field address mark. An error here causes an immediate halt, with SYNCER set in the SEQCTL register, because the FAIL bit is set.

07: This is the data marker byte. The CMPEN bit indicates that the byte must compare exactly with the immediate data byte. An error here causes an immediate halt, with SYNCER set in the SEQCTL register, because the FAIL bit is set.

08-0A: These are the first three instructions that define the sector data field. The data field is split into four control instructions so that switching between sector sizes will not require rearranging the control store data (see above). The example shows a 256 byte sector. On ECC verify commands, change BUFF to NOXFER. On compare verify commands, set BUFF, the CMPEN bit in the control byte, and RCMP in the SEQCTL register.

0B: The last instruction of the data field marks the end of the data field using the LAST bit set with BUFF. LAST must be used for BUFF or NOXFER.

0C: Data ECC field. ECC degree six with interleave factor of five is used, the count field is set to (30-1) bytes. This instruction switches the checksum shift register into check mode.

0D: The purpose of this instruction is to turn off RG, resetting any external data decoder circuits. The JMPEN bit is also set in this instruction, meaning that the next instruction to be accessed will be the one addressed by the LOOP register, in this case 00. This happens only if the SECCNT register is not zero. If SECCNT is zero, the next instruction is 12. The FAIL bit is also set, causing the checksum status to be checked, and if there was an error, the command stops, with ECCERR set in the SEQCTL register.

0E: After the last sector, the SECCNT register is zero. After instruction 0D is done, the next instruction will be this one instead of instruction 00 (LOOP register). This instruction stops the command immediately.

The CPU performs the READ SECTOR command as follows:

1. After setting up the control store and address registers as defined above, write the sector count to the SECCNT register to start the transfer.

2. Write FAULT, IDFULL, and SEQSTP true to the SISR register to clear those status bits, and to the SIMR register to enable them.

3. Wait until GINT is true with a timeout in case INDEX is not present on the interface. Check FAULT for error, and abort if true. If IDFULL is true, perform any checks on the value in the ID FIFO. If SEQSTP is true, make sure that no error occurred that must be serviced. Clear the status serviced.

6.6.4.3 Write Sector Example

ADDRESS	VALUE	COUNT	CONTROL	NOTES
00	A1	WIAM	RG,CWSEL, RTY	address mark search
01	FE	00	RG,CMPEN, RTY	ID marker byte
02	ID	02	RG, SVSEL, CMPEN	ID address
03	ID	00	RG, SVSEL	ID flag
04	CHK	01	RG, SVSEL	CRC
05	00	02	RTY	wait past splice
06	00	0B	WG, CMPEN, DAC	PLL sync
07	A1	00	WG, AM, DAC	address mark
08	F8	00	WG, DAC	data marker byte
09	BUFF	FF	WG, SVSEL,DAC	data field
0A	BUFF	FF	WG, SVSEL,DAC	data field
0B	BUFF	FF	WG, SVSEL,DAC	data field
0C	BUFF, LAST	FF	WG, SVSEL,DAC	data field
0D	CHK	1D	WG, SVSEL,DAC	ECC
0E	00	02	WG,JMPEN,DAC	write splice
0F	00	STOP	CW SEL	stop at end of cmd
START = 00, LOOP = 00, SKIP = NOT USED				

TABLE 6-39. WRITE SECTOR EXAMPLE CONTROL STORE

Line by line discussion:

00: This is the first instruction of the write sector command. The WIAM bit tells the WD10C01A to search for an ID address mark byte. The WD10C01A will not timeout, so the CPU will have to abort after it times out, if necessary. The DAC bit is zero during the ID field to select the ID checksum. The RTY bit is set so that if a read error (in this case, no sync match after AMDET true) occurs, a retry will be performed.

01: This tells the WD10C01A to verify the ID marker byte. If it does not compare exactly, an ID retry occurs (see below) because the RTY bit is set.

02: This is the ID address field. The CMPEN bit causes a compare of the ID write registers with the incoming ID field. The incoming ID is also written to the ID read registers.

03: This is the ID flag field. This instruction is separate from the ID address field because the flag byte is not compared with the ID write register. It is still written to the ID read register.

04: ID CRC field. The count field is set to (2-1) bytes for the ID CRC. This instruction switches the checksum shift register into check mode.

05: This instruction spaces over the ID field write splice to the start of the data field, protecting the ID checksum in the process. This instruction also performs a check for an ID retry. The ID compare status and checksum error status are latched and held, and the RTY bit is now set, allowing the checking of this status. An ID retry occurs if the ID

did not match, or if there was a checksum error. The checksum error is latched into the IDERR bit for future reference.

06: Data PLL sync field. This instruction is the start of the update write sector data. The WG - CMPEN combination also marks the start of the ECC calculation starting with instruction 07.

07: This is the first address mark byte for the data field. The AM bit generates AMENA with the AMC register, suppressing the clock bit in the A1 byte.

08: This is the data marker byte, and is treated as simple immediate data for write sector purposes.

09-0A: These are the first three instructions that define the sector data field. The data field is split into four control instructions so that switching between sector sizes will not require reordering the control store data (see above). The example shows a 1024 byte sector.

0B: The last instruction of the data field marks the end of the data field using the LAST bit set with BUFF.

0C: Data ECC field. ECC degree six with interleave factor of five is used, the count field is set to (30-1) bytes. The end of this instruction also stops the checksum calculation.

0D: Data field write splice. This instruction pads the end of the data field, making sure that the checksum is not lost by the drive. The JMPEN bit is also set in this instruction, meaning that the next instruction to be accessed will be the one addressed by the LOOP register, in this case 00. This happens only if the SECCNT register is not zero. If SECCNT is zero, the next control instruction is 0E.

0E: After the last sector, SECCNT is zero. After instruction 11 is done, the next instruction will be this one instead of instruction 00 (LOOP register). This instruction stops the command immediately.

The CPU performs the WRITE SECTOR command as follows:

1. After setting up the control store and address registers as defined above, write the sector count to the SECCNT register to start the transfer.

2. Write IDFULL, FAULT, and SEQSTP true to the SISR register to clear those status bits, and to the SIMR register to enable them.

3. Wait until GINT is true with a timeout in case INDEX is not present on the interface. Check FAULT for error, and abort if true. If IDFULL is true, perform any checks on the data in the ID FIFO. If SEQSTP is true, check to see if an error occurred that must be serviced. Clear the status serviced.

6.6.4.4 Read and Write Sector Example

This example makes use of the SKIP register to merge the read and write sector operations into one control store program. Selection between read and write is performed solely by changing the contents of the SKIP register. The control store example shown in the table below is divided into the ID search, read data field, and write data field sections.

ADDRESS	VALUE	COUNT	CONTROL	NOTES
00	A1	WIAM	RG,CWSEL, RTY	address mark search
01	FE	00	RG,CMPEN, RTY	ID marker byte
02	ID	02	RG, SVSEL, CMPEN	ID address
03	ID	00	RG, SVSEL	ID flag
04	CHK	01	RG, SVSEL	CRC
05	00	02	SKPEN, RTY	retry ID
08	00	00	00	wait past splice
09	A1	WDAM(1F)	RG,CWSEL,DAC,FAIL	address marker search
0A	F8	00	RG, SVSEL,DAC	data marker byte
0B	BUFF	3F	RG, SVSEL,DAC	data field
0C	BUFF	3F	RG, SVSEL,DAC	data field
0D	BUFF	3F	RG, SVSEL,DAC	data field
0E	BUFF, LAST	3F	RG,SVSEL,DAC	data field
0F	CHK	1D	RG,SVSEL,DAC	ECC
10	00	02	JMPEN,DAC, FAIL	end of sector
11	00	STOP	CW SEL	stop at end of read
14	00	0B	WG, CMPEN,DAC	PLL sync
15	A1	00	WG, AM,DAC	address mark
16	F8	00	WG, DAC	data marker byte
17	BUFF	FF	WG, SVSEL,DAC	data field
18	BUFF	FF	WG, SVSEL,DAC	data field
19	BUFF	FF	WG, SVSEL,DAC	data field
1A	BUFF, LAST	FF	WG, SVSEL,DAC	data field
1B	CHK	1D	WG, SVSEL,DAC	ECC
1C	00	02	WG,JMPEN,DAC	write splice
1D	00	STOP	CW SEL	stop at end of cmd
START = 00, LOOP = 00, SKIP = 08 for read, 14 for write				

TABLE 6-40. READ AND WRITE SECTOR EXAMPLE CONTROL STORE

Line by line discussion of Read and Write example:

00: This is the first instruction of the ID search. The WIAM bit tells the WD10C01A to search for an ID address mark byte. The WD10C01A will not timeout, so the CPU will have to abort after it times out, if necessary. The DAC bit is zero during the ID field to select the ID checksum. The RTY bit is set so that if a read error (in this case, no sync match after AMDET true) occurs, a retry will be performed.

01: This tells the WD10C01A to verify the ID marker byte. If it does not compare exactly, an ID retry occurs (see below) because the RTY bit is set.

02: This is the ID address field. The CMPEN bit causes a compare of the ID write registers with the incoming ID field. The incoming ID is also written to the ID read registers.

03: This is the ID flag field. This instruction is separate from the ID address field because the flag byte is not compared with the ID write register. It is still written to the ID read register.

04: ID CRC field. The count field is set to (2-1) bytes for the ID CRC. This instruction switches the checksum shift register into check mode.

05: This instruction performs a check for an ID retry. The ID compare status and checksum error status are latched and held, and the RTY bit is now set, allowing the checking of this status. An ID retry occurs if the ID did not match, or if there was a checksum error. The checksum error is latched into the IDERR bit for future reference. Also, the SKPEN bit is set, meaning the next address (assuming no errors occurred) will be the value in the SKIP register, which is 08 for read sector, and 14 for write sector.

READ DATA FIELD

08: This instruction spaces over the ID field write splice to the start of the data field, protecting the ID checksum in the process.

09: The WDAM bit tells the WD10C01A to look for the data address mark. The (1F) indicates the byte count that is loaded into the byte count. This value is the maximum number of byte times that the WD10C01A will look for the address mark. This prevents locking up on a subsequent data field address mark. An error here causes an immediate halt, with SYNCER set in the SEQCTL register, because the FAIL bit is set.

0A: This is the data marker byte. The CMPEN bit indicates that the byte must compare exactly with the immediate data byte. An error here causes an immediate halt, with SYNCER set in the SEQCTL register, because the FAIL bit is set.

0B-0D: These are the first three instructions that define the sector data field. The data field is split into four instructions so that switching between sector sizes will not require reordering the control store data (see above). The example shows a 256 byte sector. On ECC verify commands, change BUFF to NOXFER. On compare verify commands, set BUFF, the CMPEN bit in the control byte, and RCMP in the SEQCTL register.

0E: The last instruction of the data field marks the end of the data field using the LAST bit set with BUFF. LAST must be used for BUFF or NOXFER.

0F: Data ECC field. ECC degree six with interleave factor of five is used, the count field is set to (30-1) bytes. This instruction switches the checksum shift register into check mode.

10: The purpose of this instruction is to turn off RG, resetting any external data decoder circuits. The JMPEN bit is also set in this instruction, meaning that the next instruction to be accessed will be the one addressed by the LOOP register, in this case 00. This happens only if the SECCNT register is not zero. If SECCNT is zero, the next control instruction is 11. The FAIL bit is also set, causing the checksum status to be checked, and if there was an error, the command stops, with ECCERR set in the SEQCTL register.

11: After the last sector, the SECCNT register is zero. After instruction 10 is done, the next instruction will be this one instead of instruction 00 (LOOP register). This instruction stops the command immediately.

WRITE DATA FIELD

14: Data PLL sync field. This instruction is the start of the update write sector data. The WG - CMPEN combination also marks the start of the ECC calculation starting with instruction 15.

15: This is the first address mark byte for the data field. The AM bit generates AMENA with the AMC register, suppressing the clock bit in the A1 byte.

16: This is the data marker byte, and is treated as simple immediate data for write sector purposes.

17-19: These are the first three instructions that define the sector data field. The data field is split into four instructions so that switching between sector sizes will not require reordering the control store data (see above). The example shows a 1024 byte sector.

1A: The last instruction of the data field marks the end of the data field using the LAST bit set with BUFF.

1B: Data ECC field. ECC degree six with interleave factor of five is used, the count field is set to (30-1) bytes. The end of this instruction also stops the checksum calculation.

1C: Data field write splice. This instruction pads the end of the data field, making sure that the checksum is not lost by the drive. The JMPEN bit is also set in this instruction, meaning that the next instruction to be accessed will be the one addressed by the LOOP register, in this case 00. This happens only if the SECCNT register is not zero. If SECCNT is zero, the next control instruction is 1D.

1D: After the last sector, SECCNT is zero. After instruction 1C is done, the next instruction will be this one instead of instruction 00 (LOOP register). This instruction stops the command immediately.

The CPU performs the READ SECTOR command as follows:

1. After setting up the control store and address registers as defined above, write 06 to the SKIP register, and then write the sector count to the SECCNT register to start the transfer.

2. Write IDFULL, FAULT, and SEQSTP true to the SISR register to clear those status bits, and to the SIMR register to enable them.

3. Wait until GINT is true with a timeout in case INDEX is not present on the interface. Check FAULT for error, and abort if true. If IDFULL is true, perform any checks on the data in the ID FIFO. If SEQSTP is true, check to see if an error occurred that must be serviced. Clear the status serviced.

The CPU performs the WRITE SECTOR command as follows:

1. After setting up the control store and address registers as defined above, write 0F to the SKIP register, and then write the sector count to the SECCNT register to start the transfer.

2. Write IDFULL, FAULT, and SEQSTP true to the SISR register to clear those status bits, and to the SIMR register to enable them.

3. Wait until GINT is true with a timeout in case INDEX is not present on the interface. Check FAULT for error, and abort if true. If IDFULL is true, perform any checks on the data in the ID FIFO. If SEQSTP is true, check to see if an error occurred that must be serviced. Clear the status serviced.

6.6.5 ID Retry and Error Conditions

The WD10C01A manages errors by using the RTY and FAIL bits in the control store error control byte. These bits tell the WD10C01A when to check its internal error status bits, and what to do about the error. All internal status bits in the WD10C01A are held in their error state until reset by a retry, or by writing to the START register.

If an error occurs when or before the RTY bit is set, an ID retry is performed. The LOOP register address will be loaded into the control store address register, restarting the sector. Also, the RG signal is switched false for one whole byte time to reset external decoders/PLLs. If the control store instruction at the LOOP address does not specify that RG is true, then it will stay false.

If an error occurs when or before the FAIL bit is set, an immediate abort of the command occurs when FAIL goes true. The appropriate error status bit(s) are set, and the sequencer stops.

The following list summarizes the read error conditions of the WD10C01A:

1. When executing a WIAM or WDAM operation, and the AMDET signal goes true, but the sync byte does not match within the sixteen bit timeout period. SYNCER will be set true if a halt is commanded (FAIL set).

2. When searching for a simple byte sync byte, and the count in the control store count field is exhausted. SYNCER will be set true if a halt is commanded (FAIL set).

3. Once byte sync is established, any read compare operations (CMPEN bit set) that fail. In the case of immediate data, this handles the ST412 second address mark byte (FE or F8) and the SYNCER bit will be true if a halt is commanded (FAIL set). In all other cases (i.e., buffer or ID data), the CMPERR bit is be true if a halt is commanded (FAIL set).

4. Any checksum error. This error status is retained by the IDERR bit in the SEQCTL register if a retry is commanded (RTY set) until the next ID field starts. The ECCERR bit is set if an abort was commanded (FAIL set). In this case, the checksum shift register contains the correction syndrome. The register ECCS shows the additional ECC status.

5. When executing a WDAM operation, if the AMDET signal does not go true, and the count in the control store count field is exhausted. SYNCER will be set true if a halt is commanded (FAIL set).

6.6.6 Error Recovery

When a read error occurs that causes a halt of the command before it is completed, certain steps must be performed to recover from that error. When the sequencer stops, the following conditions are in effect:

1. The SECCNT register contains a remainder. If the error is an ID error or a sync error (SYNCER set), this is the number of sectors to read including the one in error. Any other error gives the number of sectors to read after the one in error.

2. The ID set registers are set to an ID field. If the error is a sync error, this is the sector address of the one in error. Any other error gives the sector address following the one in error. In general, any error that is detected after the data field transfer starts (DXF went true, see the SEQSTS register) will cause SECCNT and the ID write registers to be set for the sector following the one in error.

3. The SEQCTL read bits define the error. If ECCERR is set, then the error is a checksum error. If CMPERR is set, the error is a verify or second address mark byte error. If SYNCER is set, the error is a byte sync error.

4. The internal checksum register may contain a correction syndrome, depending on the type of error.

The SECCNT contains a non-zero remainder. SECCNT is inhibited from restarting the sequencer by the latched error status. In order to restart for retry or continuing, the START register must be written with the starting control store address. This immediately clears any error status (SEQCTL read bits), and if the SECCNT register is still non-zero, will cause the sequencer to start.

6.6.7 Error Correction

The Reed-Solomon ECC implemented in the WD10C01A can correct up to 2 error bytes (degree 5) and up to 3 error bytes (degree 6) per interleave. Depending on the interleave factor being used, it can correct up to 30 bytes of error within a sector.

During read operations, the device produces composite syndromes. These syndromes are used by the error correction routine (microprocessor) to calculate the error location and error value. The microprocessor responses to the SEQSTP interrupt by first reading the SISR and the ECCS registers to determine the type of failure. If the EERR bit in the ECCS is set, the microprocessor determines which interleaves are in error by reading the status bits I0E-I4E in the ECCS. The microprocessor then start reading the SPORT register to transfer the syndromes. The WD10C01A automatically transfer the syndromes from the lowest interleave number. The highest byte is transferred first. The microprocessor has to read the SPORT register five times for degree 5 and six times for degree 6 to complete the syndrome transfer of an interleave. After all syndromes of the interleave are transferred, the microprocessor must reset the corresponding error status bit in the ECCS by writing a 0 into that bit. The microprocessor can then continue reading the syndromes of the next interleave.

7.0 DC ELECTRICAL SPECIFICATIONS

7.1 MAXIMUM RATINGS

Ambient temperature	0°C to 70°C
Storage temperature	-65° C to 150° C
Voltage on any pin with respect to V_{SS}	-0.3 to V_{DD}+0.3 Volts
Voltage on V_{DD} with respect to V_{SS}	7 Volts
Leakage current	±10 µA
Power dissipation	1000 mW at X1=32 MHz, RRCLK=27 MHz, 0°C, all outputs open
Input Static Discharge Protection	2000 V pin to pin

NOTE

Stresses above those listed under Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational section of the specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect product reliability.

7.2 STANDARD TEST CONDITIONS

The characteristics below apply for the following standard test conditions, unless otherwise noted. All voltages are referenced to V_{SS} (0V Ground). Positive current flows into the referenced pin.

Operating temperature range	0° to 70° C
V_{DD} supply voltage with respect to V_{SS}	+5 Volts ± 0.5 V
V_{SS}	0 Volts
Latch-up current (min)	±40 mA
Operating humidity range	20 to 95%
X1 input operating frequency with crystal	32 MHz(max) 8 MHz (min)
X1 input operating frequency with TTL source	25 MHz(max) --- MHz (min)

20

7.3 DC CHARACTERISTICS

SYMBOL	PARAMETER	MIN	MAX	UNITS	CONDITIONS
V_{IH}	Input High Voltage	2.0	---	V	V_{DD} = 5V ±5%
V_{IL}	Input Low Voltage	---	0.8	V	V_{DD} = 5V ±5%
I_{IH}	Input High Current	---	600	µA	V_{IL} = 0.8V
I_{IL}	Input Low Current	---	-600	µA	V_{IH} = 2.0V
V_{OH}	Output High Voltage	2.4	---	V	*, I_{OH} = -400 µA
V_{OL}	Output Low Voltage	---	0.40	V	*, I_{OL} = 2 mA
V_{OH}	Output High Voltage	V_{DD}-0.5	---	V	**,I_{OH} = -800 µA
V_{OL}	Output Low Voltage	---	0.40	V	**, I_{OL} = 4 mA
V_{OH}	Output High Voltage	2.4	---	V	***,I_{OH} = -2.5 mA
V_{OL}	Output Low Voltage	---	0.40	V	****, I_{OL} =6 mA
C_I	Input Capacitance	---	10	pF	all inputs
C_O	Output Capacitance	---	50	pF	All outputs except: D0-D7, BMD0-BMD7,BMP, OSC, and CPUCLK
C_O	Output Capacitance	---	100	pF	Outputs D0-D7,BMD0-BMD7, and BMP
C_O	Output Capacitance	---	100	pF	Outputs OSC and CPUCLK
I_{CC}	Supply Current	---	200	mA	60 mA (typical); X1=32MHz, RRCLK=27MHz, 0°C, all outputs open, V_{DD}=5V

TABLE 7-1. DC CHARACTERISTICS

NOTES:

Ta = 0°C (32°F) to 70°C (158°F),
V_{DD} = +5V ±5%

* Output Voltages (all outputs except X2, CPUCLK, OSC, D0-D7), see †

** Output Voltages (CPUCLK and OSC only), see†

***Output Voltages (D0-D7), see †

****Output Voltages (D0-D7, $\overline{INT}$ Logic 0)

† Even under worst case AC transient switching conditions V_{OL} = 0.8V shall not be exceeded on any output pin at any time.

8.0 AC OPERATING CHARACTERISTICS

The following notes apply to all of the parameters presented in this section:

1. All unit are in nanoseconds

2. These timing relationships assume the maximum capacitive loading for both inputs and outputs, V_{DD} = 4.50 volts to 5.50 volts.

3. Temperature = 0°C to 70°C.

4. All timing is measured between 0.8 volts logic low and 2.0 volts logic high, unless otherwise noted.

8.1 OSC AND CPUCLK TIMING

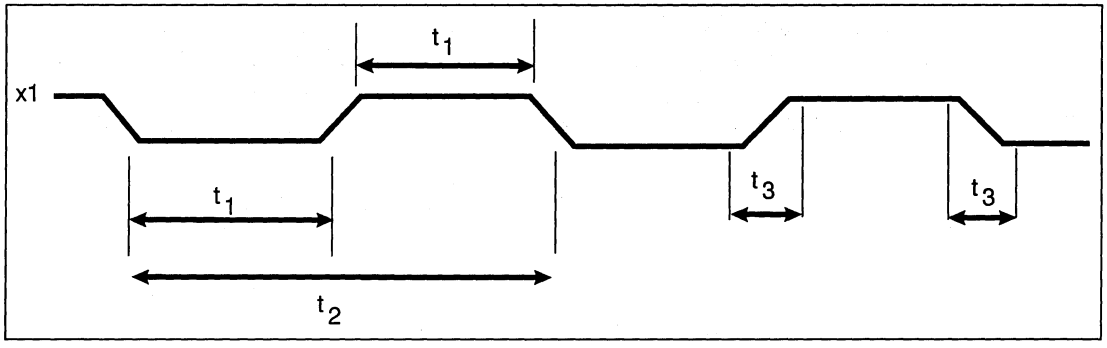

FIGURE 8-1. TTL SOURCE X1 CLOCK INPUT

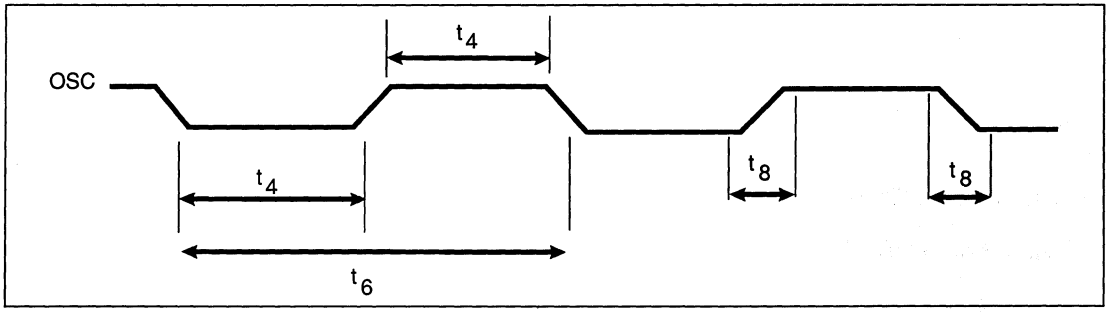

FIGURE 8-2. OSC OUTPUT

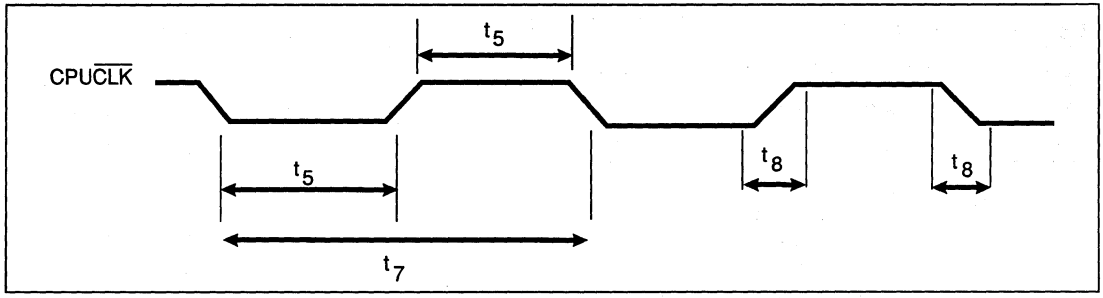

FIGURE 8-3. CPUCLK OUTPUT

The following table summarizes the relationship between the clock at X1 and the resultant outputs at OSC and CPUCLK. OSCDIV and CPUDIV are control bits in the RESET register that determine how the X1 clock is divided to produce OSC and CPUCLK. See the programming section for more information.

OSCDIV	CLKDIV	OSC	CPUCLK
0	0	X1/2	X1/6
0	1	X1/2	X1/4
1	0	X1/1	X1/3
1	1	X1/1	X1/2

No.	DESCRIPTION	MIN	MAX	UNITS
t_1	TTL source X1 high or low(*1)	13		ns
t_2	TTL source X1 cycle time	40		ns
t_3	TTL source X1 rise or fall time (*1)		5	ns
t_4	OSC high or low when: (*2)		---	---
	X1/1 crystal (*3)	10	---	ns
	X1/2 crystal (*3)	27	33	ns
	X1/1 TTL source (*5)	16	24	ns
	X1/2 TTL source (*4)	36	44	ns
t_5	CPUCLK high or low when: (*2)	---	---	---
	X1/2 crystal (*3)	27	33	ns
	X1/3 crystal (*3)	36	54	ns
	X1/4 crystal (*3)	54	66	ns
	X1/6 crystal (*3)	81	99	ns
	X1/2 TTL source (*4)	36	44	ns
	X1/3 TTL source (*5)	48	72	ns
	X1/4 TTL source (*4)	72	88	ns
	X1/6 TTL source (*4)	108	132	ns
t_6	OSC cycle time when:	---	---	---
	X1/1 crystal (*3)	30	125	ns
	X1/2 crystal (*3)	60	250	ns
	X1/1 TTL source (*4)	40	---	ns
	X1/2 TTL source (*4)	80	---	ns
t_7	CPUCLK cycle time when:	---	---	---
	X1/2 crystal (*3)	60	250	ns
	X1/3 crystal (*3)	90	375	ns
	X1/4 crystal (*3)	120	500	ns
	X1/6 crystal (*3)	180	750	ns
	X1/2 TTL source (*4)	80	---	ns
	X1/3 TTL source (*4)	120	---	ns
	X1/4 TTL source (*4)	160	---	ns
	X1/6 TTL source (*4)	240	---	ns
t_8	CPUCLK and OSC rise or fall time (*2)	---	5	ns

TABLE 8-1. OSC AND CPUCLK TIMING PARAMETERS

NOTES:

*1)Times are measured relative to V_{IH} and V_{IL}.

*2)High and low times are measured relative to the midpoints between V_{OL} and V_{OH}. Rise and fall times are measured between V_{OH} and V_{OL}.

*3)Assumes 33.3 MHz crystal across X1 and X2 for min times, 8.0 MHz crystal for max times.

*4)Assumes 25.0 MHz TTL source to X1.

*5)Assumes 25.0 MHz TTL source to X1, 50/50 duty cycle.

8.2 CPU INTERFACE TIMING

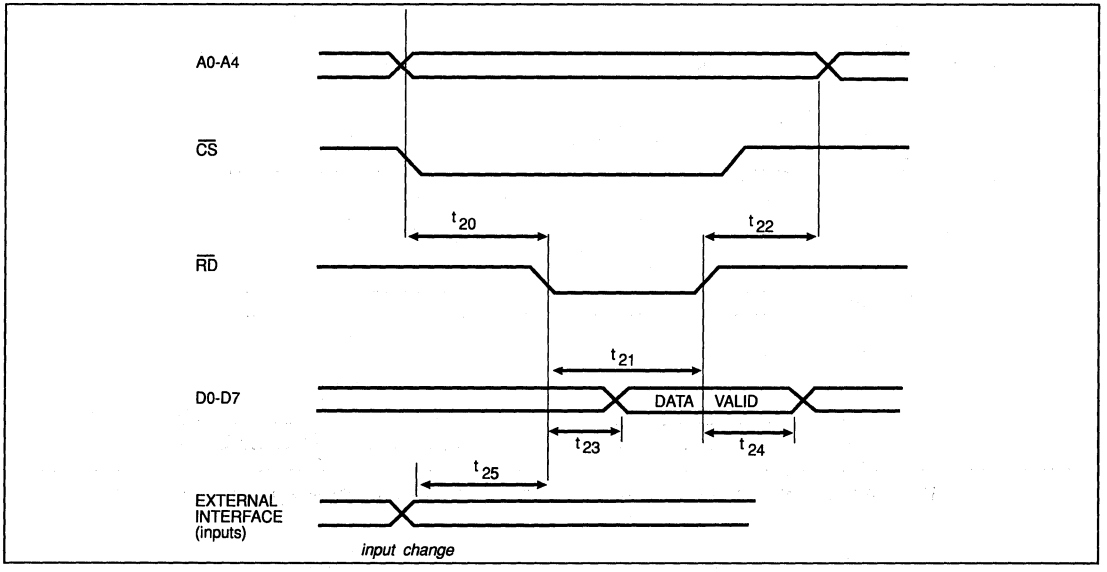

FIGURE 8-4. MICROPROCESSOR RD* TIMING (RD* CONTROLLED)

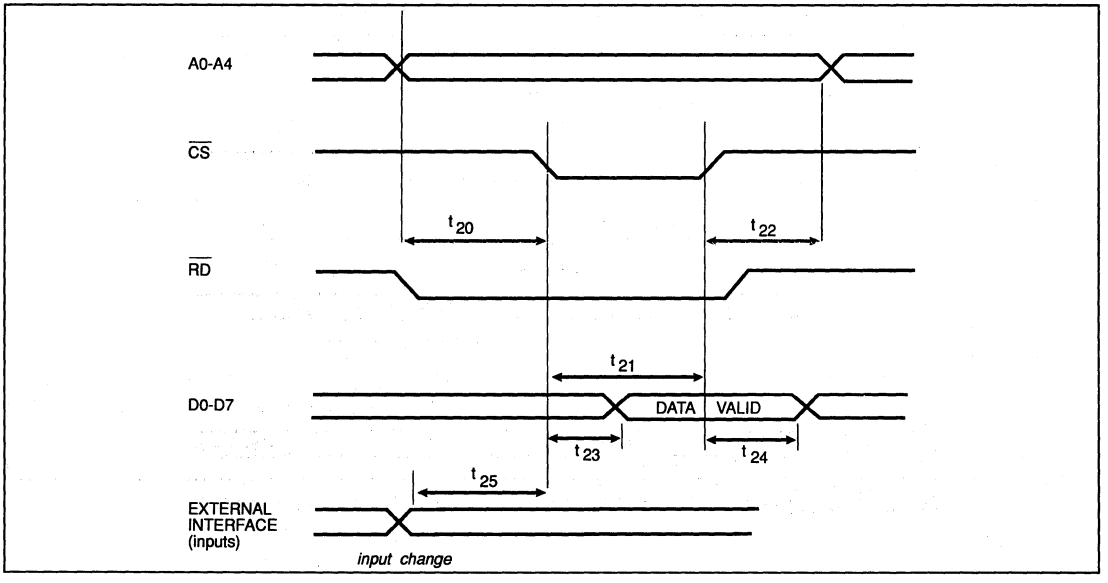

FIGURE 8-5. MICROPROCESSOR RD* TIMING (CS* CONTROLLED)

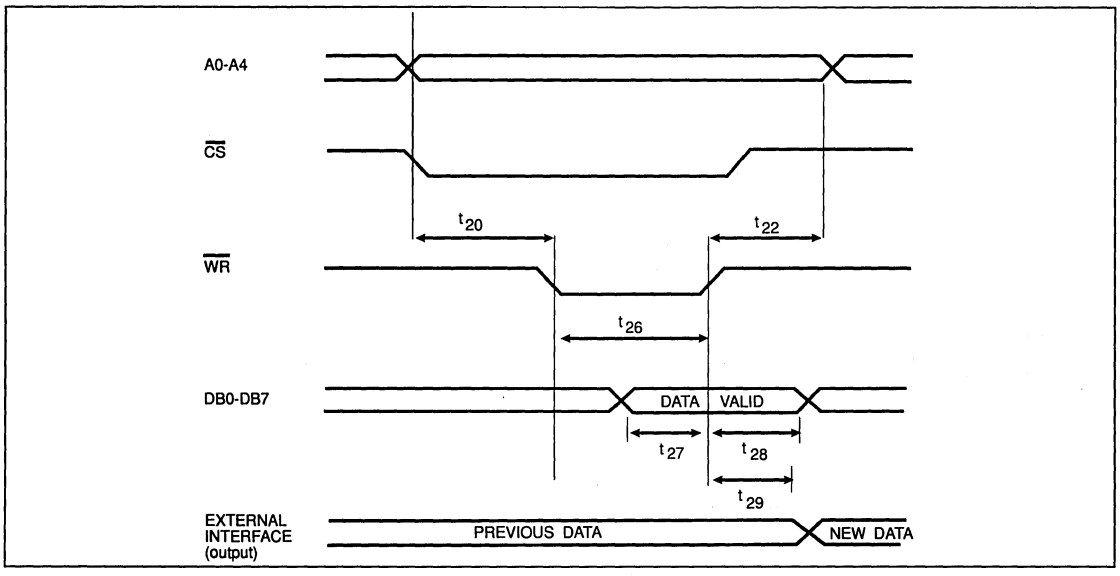

FIGURE 8-6. MICROPROCESSOR WR* TIMING (WR* CONTROLLED)

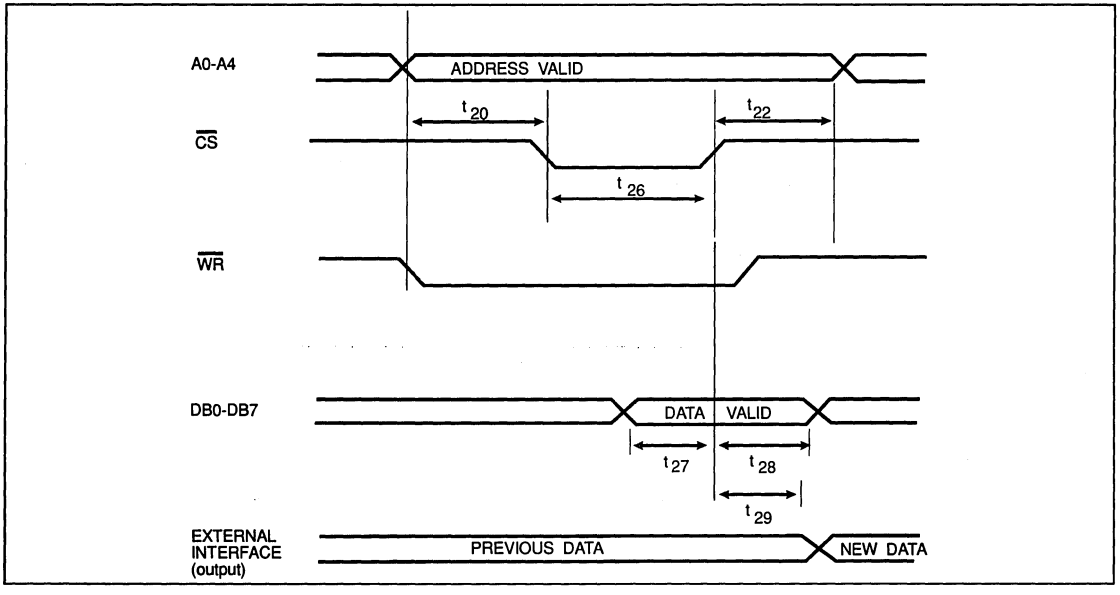

FIGURE 8-7. MICROPROCESSOR WR* TIMING (CS* CONTROLLED)

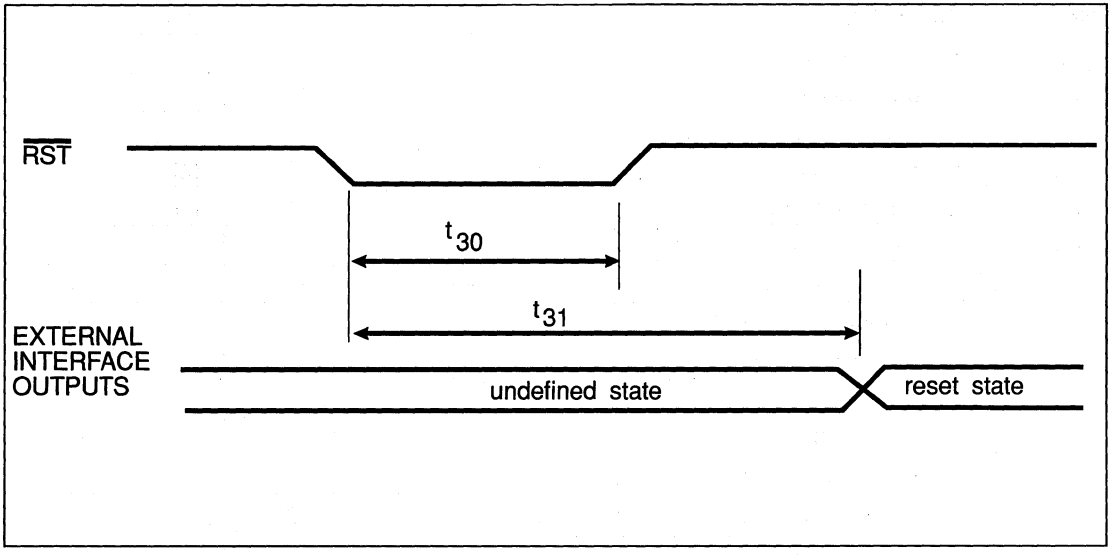

FIGURE 8-8. RESET TIMING

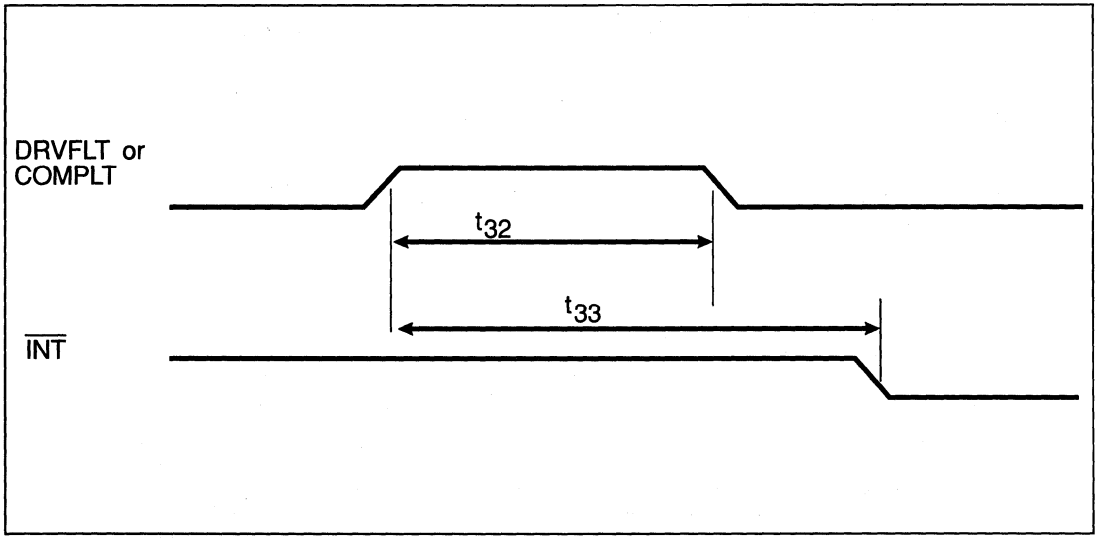

FIGURE 8-9. EXTERNALLY GENERATED INTERRUPT TIMING

No.	DESCRIPTION	MIN	MAX	UNITS
t_{20}	address valid to $\overline{RE}$ or $\overline{WE}$ (*3)	20		ns
t_{21}	$\overline{RE}$ pulse width (*3)	100		ns
t_{22}	$\overline{RE}$ or $\overline{WE}$ to address change (*3)	0		ns
t_{23}	$\overline{RE}$ true to data valid (*3)	---	95	ns
t_{24}	$\overline{RE}$ false to data hold (*3)	20	60	ns
t_{25}	input port setup to $\overline{RE}$ true (*1)(*3)	80	---	ns
t_{26}	$\overline{WE}$ pulse width (*3)	100	---	ns
t_{27}	data setup to $\overline{WE}$ false (*3)	80		ns
t_{28}	$\overline{WE}$ false to data hold (*3)	0		ns
t_{29}	$\overline{WE}$ false to output change (*2)(*3)	---	80	ns
t_{30}	$\overline{RST}$ pulse width	100	---	ns
t_{31}	$\overline{RST}$ true to stable outputs	---	150	ns
t_{32}	DRVFLT or COMPLT pulse width	100	---	ns
t_{33}	DRVFLT or COMPLT high to $\overline{INT}$ low	---	150	ns

TABLE 8-2. CPU INTERFACE TIMING PARAMETERS

NOTES:

*1) Inputs are: PZ0-5, PY0-3 when defined as inputs, and AMDET when being used as a simple input pin.

*2) Outputs are: PX0-7, and PY0-3 when defined as outputs.

*3) $\overline{RE}$ = $\overline{RD}$ or $\overline{CS}$;
$\overline{WE}$ = $\overline{WR}$ or $\overline{CS}$

8.3 BUFFER INTERFACE TIMING

(Data is coming out of the WD10C01A)
REQTIM=1 in RESET register. (REQA true when internal data register is full.)

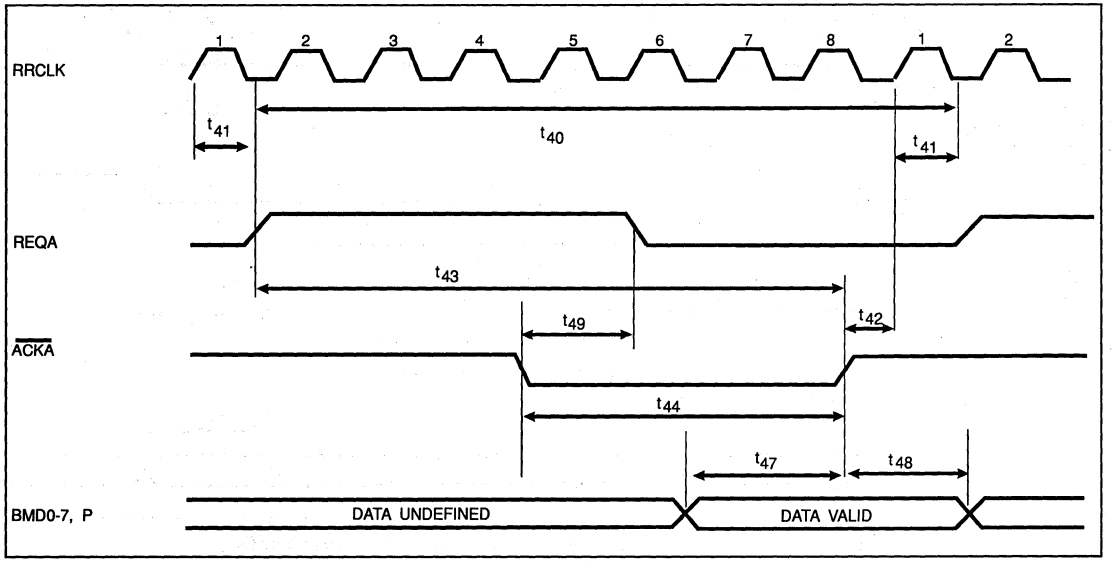

FIGURE 8-10. ASYNCHRONOUS MODE DATA BUS WRITE TIMING (1)

REQTIM=0 in RESET register. (REQA true one bit time before internal data register is full. Note that the cycles overlap by one bit time.)

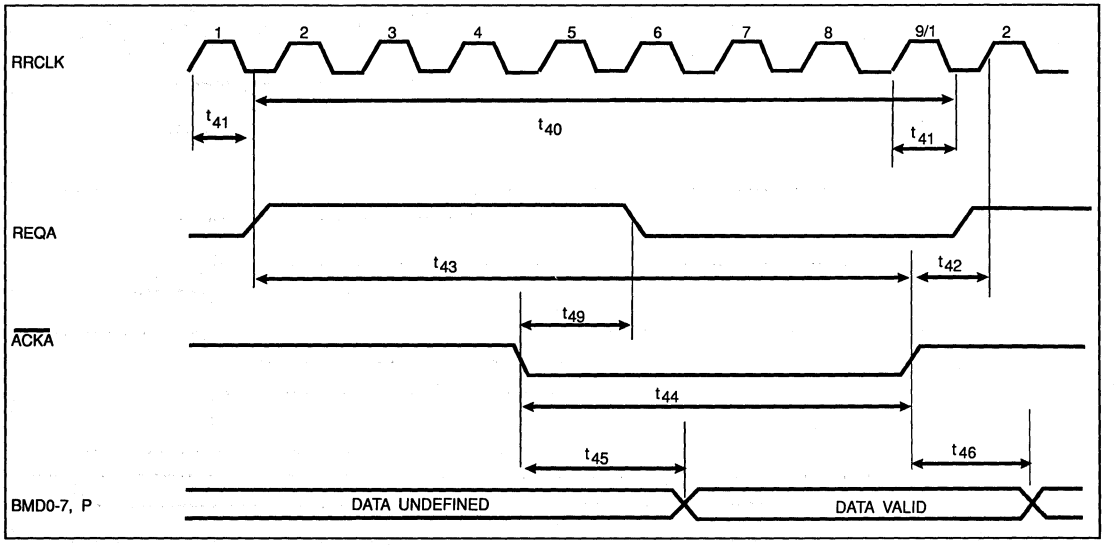

FIGURE 8-11. ASYNCHRONOUS MODE DATA BUS READ TIMING (0)

(Data is going into the WD10C01A)
REQTIM=1 in RESET register. (REQA true when internal data register is empty.)

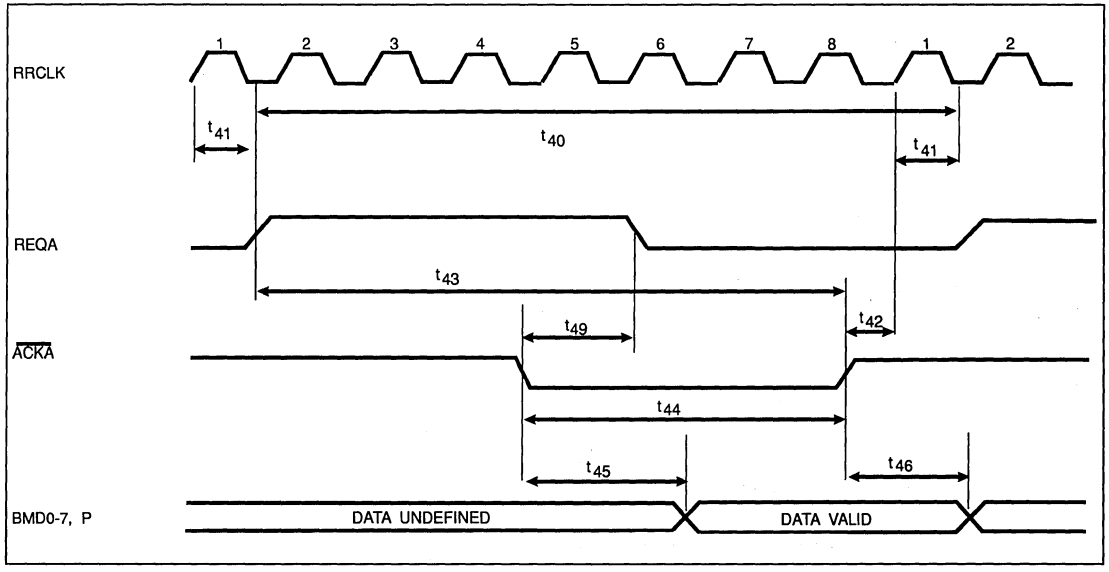

FIGURE 8-12. ASYNCHRONOUS MODE DATA BUS READ TIMING (1)

REQTIM=0 in RESET register. (REQA true one bit time before internal data register is empty. Note that the cycles overlap by one bit time.)

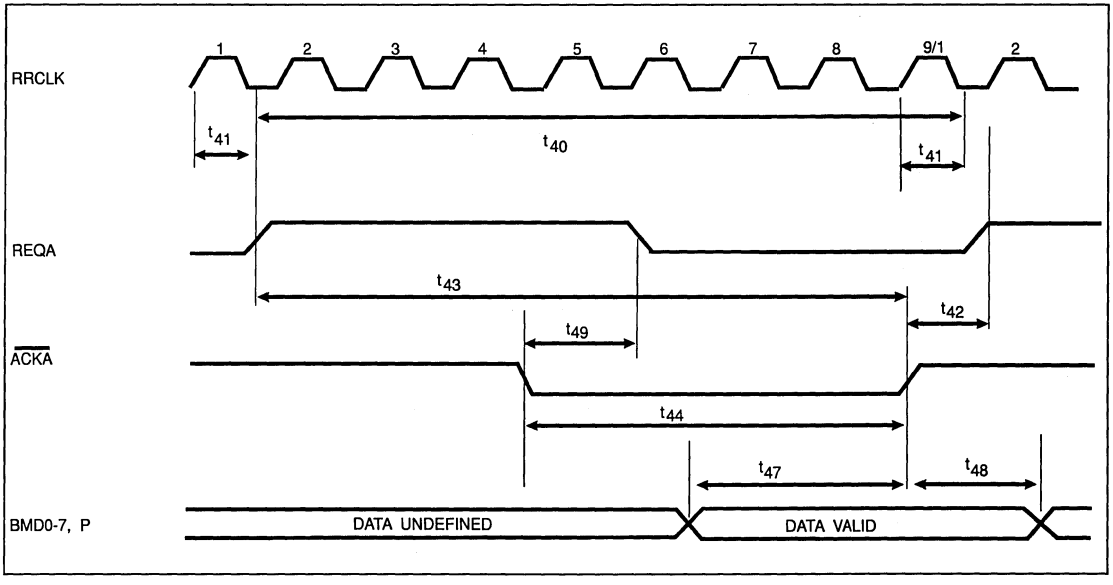

FIGURE 8-13. ASYNCHRONOUS MODE DATA BUS WRITE TIMING (0)

No.	DESCRIPTION	MIN	MAX	UNITS
t_{40}	DMA (REQA) cycle time (*1)		$8*T_{cyc}$	ns
t_{41}	RRCLK true to REQA true		40	ns
t_{42}	$\overline{ACKA}$ false to RRCLK true	20		ns
t_{43}	REQA true to $\overline{ACKA}$ false: (*1) REQTIM=0 REQTIM=1		--- $9*T_{cyc}-t_{41}-t_{42}$ $8*T_{cyc}-t_{41}-t_{42}$	ns
t_{44}	$\overline{ACKA}$ active low	100	---	ns
t_{45}	$\overline{ACKA}$ true to data valid		60	ns
t_{46}	$\overline{ACKA}$ false to data hold	10	60	ns
t_{47}	data setup to ACKA* false	35		ns
t_{48}	$\overline{ACKA}$ false to data hold	5		ns
t_{49}	$\overline{ACKA}$ true to REQA false		35	ns

TABLE 8-3. BUFFER INTERFACE TIMING PARAMETERS

NOTES:

*1)T_{cyc} is the RRCLK cycle time used.

8.4 SERIAL DATA TIMING

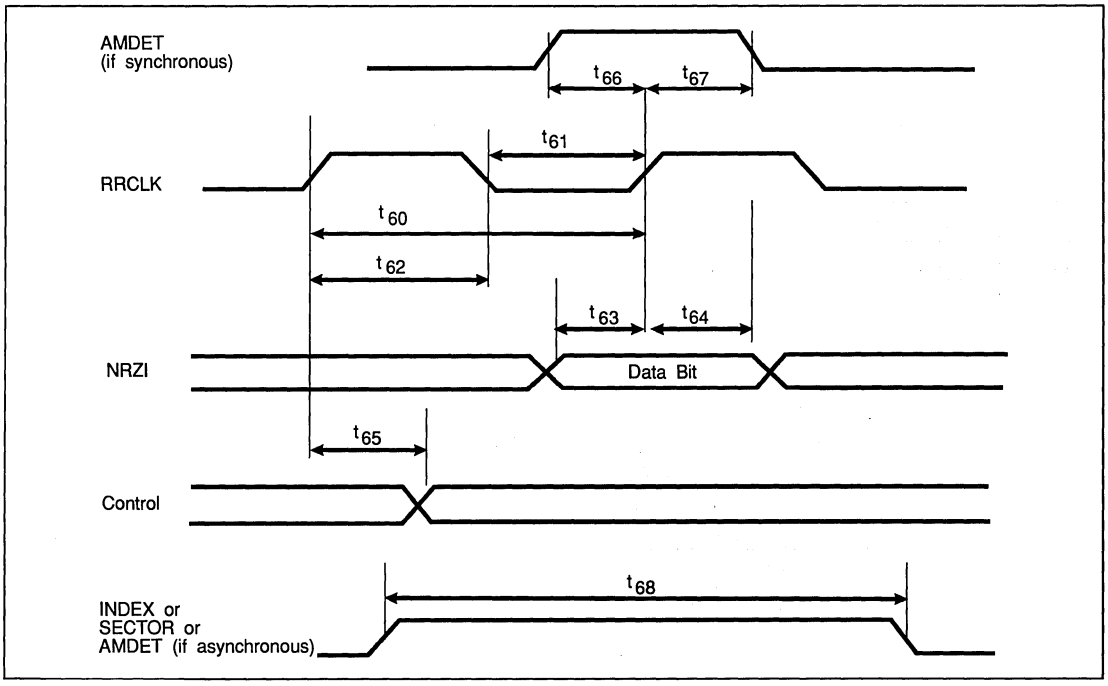

FIGURE 8-14. NRZ DATA INPUT TIMING

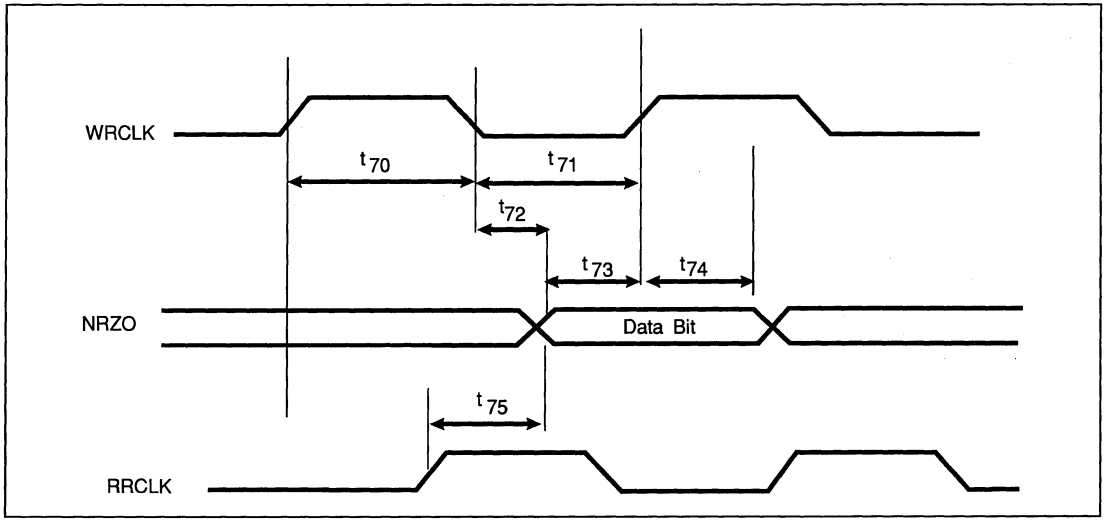

FIGURE 8-15. NRZ DATA OUTPUT TIMING

No.	DESCRIPTION	MIN	MAX	UNITS
t_{60}	RRCLK cycle time	37		ns
t_{61}	RRCLK low (*1)	14		ns
t_{62}	RRCLK high (*1)	14		ns
t_{63}	NRZI setup to RRCLK high	10		ns
t_{64}	RRCLK high to NRZI hold	10		ns
t_{65}	RRCLK high to new control out (*3)		30	ns
t_{66}	AMDET setup to RRCLK high (*4)	10		ns
t_{67}	RRCLK high to AMDET hold (*4)	10		ns
t_{68}	asynchronous input width	$2*T_{cyc}$		ns
t_{70}	WRCLK high (*2) (*6) (*2) (*6) (*7)	T_{rlo}-6.0 T_{rlo}-5.0		ns
t_{71}	WRCLK low (*2) (*6) (*2) (*6) (*7)	T_{rhi}-6.0 T_{rhi}-5.0		ns
t_{72}	WRCLK low to NRZO change(*6) (*6) (*7)	-3.5 -2.75	3.5 2.75	ns
t_{73}	NRZO setup to WRCLK high(*6) (*6) (*7)	T_{rhi}-9.5 T_{rhi}-8.75		ns
t_{74}	WRCLK high to NRZO hold (*6) (*6) (*7)	T_{rlo}-9.5 T_{rlo}-8.75		ns
t_{75}	RRCLK high to NRZO change		30	ns

TABLE 8-4. SERIAL DATA TIMING PARAMETERS

NOTES:

*1)High and low times measured relative to V_{IH} and V_{IL}.

*2)T_{rlo} and T_{rhi} are the clock low and clock high (respectively) for the RRCLK input used. T_{cyc} is RRCLK cycle time used.

*3)Control outputs are: SEQOUT, WG, RG, and AMENA.

*4)When AMDET is supplied from a synchronous source.

*5)Asynchronous inputs are: INDEX, SECTOR, and AMDET when it is supplied from an asynchronous source.

*6)Where the RRCLK input is driven from 0.4V (V_{IL}) to 2.4V (V_{IH}).

*7)The specification with reduced load capacitance of 25 pF.

A.0 WD10C01A PROGRAMMER'S BENCH REFERENCE (PBR)

ADDR	NAME	DIR	SIZE
00	SRESET	W	6-0
01	SISR	R/W	7-0
02	SIMR	R/W	7-0
03	SEQSTS/PYC	R/W	7-0/ 3-0
04	CSERR	R/W	3-0
05	CSCTL	R/W	7-0
06	CSVAL	R/W	7-2; 7-0
07	CSCNT	R/W	7-4, 0;7-0
08	PORTX	R/W	7-0
09	PORTY	R/W	3-0
0A	PORTZ/AMC	R/W	5-0/ 7-0
0B	SEQCTL	R/W	5-0
0C	START	R/W	4-0
0D	LOOP	W	4-0
0E	ECCCTL	W	5-1
0F	SECCNT	R/W	7-0
10	ECCP	R/W	3-0
11	ECCS	R/W	4-0
12	SPORT	R	7-0
13	Reserved		7-0
14	Reserved		7-0
15	Reserved		7-0
16	do not use - test only		
17	SKIP	W	4-0
18	ID0	R/W	7-0
19	ID1	R/W	7-0
1A	ID2	R/W	7-0
1B	ID3	R/W	7-0
1C	ID4	R/W	7-0
1D	ID5	R/W	7-0
1E	ID6	R/W	7-0
1F	ID7	R/W	7-0

TRUE = 1 FOR ALL BITS

A.1 ADDRESS BIT TABLES

The following is a set of bit tables.

SRESET(00)	
Bit	**Write**
7	CLKDIV
6	OSCDIV
5	
4	
3	
2	REQTIM
1	ID3$4
0	SRST

SISR(01) AND SIMR(02)	
Bit	**Read/Write**
7	GINT
6	IDFULL
5	DXFER
4	COMPLT
3	SEQSTP
2	SECEND
1	SM$IX
0	FAULT

SEQSTS(03)	
Bit	**Read**
7	DATFLD
6	ECCEN
5	LAST
4	ID
3	CHK
2	WAIT
1	AMDET
0	SEQOUT

PYC(03)	
Bit	**Write**
7	
6	
5	
4	
3	PY3OUT
2	PY2OUT
1	PY1OUT
0	PY0OUT

CSVAL(06)	
Bit	**Read/Write**
7	BUFF
6	NOXFER
5	LAST
4	ID
3	CHK
2	
1	
0	

CSERR(04)	
Bit	**Read/Write**
7	
6	
5	
4	
3	FAIL
2	RTY
1	DAC
0	SEQOUT

CSCNT(07)	
Bit	**Read/Write**
7	WDAM (COUNT)
6	WIAM
5	WIX
4	WSM
3	
2	
1	
0	STOP

CSCTL(05)	
Bit	**Write**
7	SVSEL
6	CWSEL
5	WG
4	RG
3	AM
2	CMPEN
1	SKPEN
0	JMPEN

SEQCTL(0B)		
Bit	**Read**	**Write**
7		RGERLY
6		
5	ECCERR	
4	IDERR	
3	PTYERR	IXMASK
2	SYNCER	SMMASK
1	CMPERR	RCMP
0		KILL

ECCCTL(0E)	
Bit	**Write**
7	
6	
5	ECCCLR
4	CRCSET
3	CRCNIT
2	IGNERR
1	DISPTY
0	

ECCS(11)		
Bit	**Read**	**Write**
7		
6	EERR	
5	CERR	
4	I4E	I4E
3	I3E	I3E
2	I2E	I2E
1	I1E	I1E
0	I0E	I0E

ECCP(10)	
Bit	**Read/Write**
7	
6	
5	
4	
3	SYNCCRC
2	SYNCECC
1	DEG6/5
0	IFS5/3

B.0 RESET CONDITIONS

The following list defines what is reset when $\overline{RST}$ is asserted on the WD10C01A, or the CPU sets the internal reset bit (SRST in SRESET register):

- SRST bit in SRESET is left set, and must be cleared by the CPU to take the WD10C01A out of the reset state
- interrupts are disabled
- PY0-3 are set to input
- BMD0-7,P are disabled
- OSC is set to X1/2
- CPUCLK is set to X1/6
- command sequencer stops

The following CPU registers are reset to zero:

- PORT X
- PORT Y CONFIGURATION
- SECTOR COUNTER
- INTERRUPT MASK REGISTER
- SRESET REGISTER (except SRST)
- ECCP except SYNCCRC and SYNCECC, which are initialized to one's
- ECCS
- SPORT
- KILL and RGERLY bits in SEQCTL register

The following outputs are reset to zero:

- PX0-7
- SEQOUT
- AMENA
- RG
- WG
- NRZO
- REQA

The following error status bits are reset to zero:

- IDERR
- PTYERR

C.0 CRYSTAL OSCILLATOR APPLICATIONS

For applications that use the internal oscillator capability of the WD10C01A, a series resonant crystal must be used. This crystal must meet the following internal specifications:

CS = 7 pf MAX

RS = 30 ohms MAX

The oscillator also requires bypass capacitors, as shown in the following diagram:The following table lists values for C1 and C2 for several typical crystal frequencies. The capacitor tolerances are ±10%. Values for intermediate frequencies (not listed in the table) may be extrapolated.

FREQ (MHz)	C1 (pf)	C2 (pf)
8	180	100
10	180	68
12	150	47
14	120	56
16	82	56
20	82	33
24	56	27
25	56	22
30	39	12
32	33	12

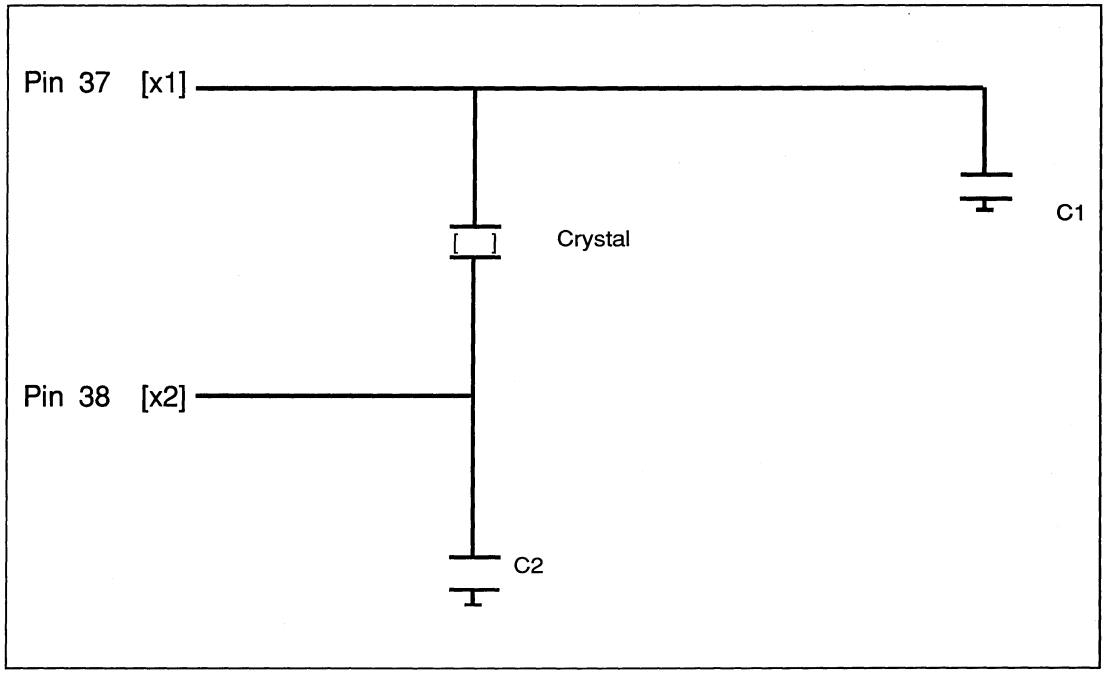

FIGURE C-1. OSCILLATOR WITH CAPACITORS

D.0 PIN/SIGNAL SUMMARY

PIN	SIGNAL	I/O	FUNCTION
1	BMD7	I/O	Buffer
2	BMD6	I/O	Buffer
3	BMD5	I/O	Buffer
4	BMD4	I/O	Buffer
5	BMD3	I/O	Buffer
6	BMD2	I/O	Buffer
7	BMD1	I/O	Buffer
8	BMD0	I/O	Buffer
9	VSS	I	Ground
10	$\overline{CS}$	I	CPU Interface
11	DB0	I/O	CPU Interface
12	DB0	I/O	CPU Interface
13	DB0	I/O	CPU Interface
14	DB0	I/O	CPU Interface
15	DB0	I/O	CPU Interface
16	DB0	I/O	CPU Interface
17	DB0	I/O	CPU Interface
18	DB0	I/O	CPU Interface
19	$\overline{RD}$	I	CPU Interface
20	$\overline{WR}$	I	CPU Interface
21	A4	I	CPU Interface
22	A3	I	CPU Interface
23	A2	I	CPU Interface
24	A1	I	CPU Interface
25	A0	I	CPU Interface
26	VDD	I	+5 Volts
27	$\overline{INT}$	O	CPU Interface
28	SECTOR	I	Disk Control
29	INDEX	I	Disk Control
30	AMDET	I	Disk Data
31	AMENA	O	Disk Data
32	SEQOUT	O	Disk Data
33	OSC	O	CPU Interface
34	DRVFLT	I	Disk Control
35	COMPLT	I	Disk Control

PIN	SIGNAL	I/O	FUNCTION
36	CPUCLK	O	CPU Interface
37	X1	I	CPU Interface
38	X2	O	CPU Interface
39	PZ5	I	Disk Control
40	PZ4	I	Disk Control
41	PZ3	I	Disk Control
42	PZ2	I	Disk Control
43	VSS	I	Ground
44	PZ1	I	Disk Control
45	PZ0	I	Disk Control
46	PY3	I,O	Disk Control
47	PY2	I,O	Disk Control
48	PY1	I,O	Disk Control
49	PY0	I,O	Disk Control
50	PX7	O	Disk Control
51	PX6	O	Disk Control
52	PX5	O	Disk Control
53	PX4	O	Disk Control
54	PX3	O	Disk Control
55	PX2	O	Disk Control
56	PX1	O	Disk Control
57	PX0	O	Disk Control
58	WG	O	Disk Data
59	RG	O	Disk Data
60	VDD	I	+5 Volts
61	RST*	I	CPU Interface
62	WRCLK	O	Disk Data
63	NRZO	O	Disk Data
64	RRCLK	I	Disk Data
65	NRZI	I	Disk Data
66	REQA	O	Buffer
67	$\overline{ACKA}$	I	Buffer
68	DBP	I/O	Buffer

E.0 DIFFERENCES BETWEEN WD10C00 AND WD10C01A

E.1 ERROR CORRECTION AND DETECTION

The computer-generated-code ECC in WD10C00 was replaced with the Reed-Solomon ECC in WD10C01A. Six registers (ECC[0:5]) which define the masks for ECC polynomials in WD10C00 were taken out. The new ECCP, ECCS, SPORT registers facilitate the configuration parameters, operation status and the syndrome access for the new RS encoder/decoder.

E.2 SRESET REGISTER

The control bits IDCHK, DCHK, ECCSIZ bits of the WD10C00 are removed. By default, data field is covered by RS-ECC, and ID Field is covered by CCITT-CRC with $g(x)= X^{16} + X^{12} + X^5 + 1$.

E.3 ECCCTL REGISTER

The ECCINL, ECCSHT, ECCINM bits of the WD10C00 are removed. The control bit ECCSET is renamed to CRCSET, bit ECCNIT is renamed to CRCNIT. These bits affect only the CRC shift registers, the RS-ECC shift registers are be default reset to zero. The control bit DISCHK of WD10C00 is changed into IGNERR. Its function is still the same.

E.4 TIMING

The timing t_{47} (data setup time on the BMD bus before ACKA goes inactive) is changed from 30 ns to 35 ns; t_{70}, t_{71}, t_{72}, t_{73} and t_{74} have additional specifications with reduced loading capacitance of 25 pF.

E.5 PIN NAME

The name of the microprocessor data bus and the DMA data bus are changed to reflect the name on the circuit diagrams. The pin order and the functions are exactly the same as before.

WD10C00	WD10C01A
DB0-DB7	BMD0-BMD7
DBP	BMDP
D0-D7	DB0-DB7

E.6 PARITY ERROR HANDLING

WD10C00 will latch the parity error condition into the SEQCTL register and the operation will continue normally. WD10C01A will stop at the end of the current CS instruction if the FAIL bit is set to one. The WG output is also deasserted.

E.7 SEQCTL REGISTER

Bit 6 and bit 0 in the SEQCTL register are unused in the WD10C00 and in the WD10C01A. These bits are read-only type and are set to one in the WD10C00; in the WD10C01A, these bits are set to zero.