XYLOGICS 732 Disk Controller User's Manual

SECTION 1: SPECIFICATIONS

1.0 GENERAL

The DCE is a multifunction VME bus interface board that consists of two separate controllers. One controller is a standard 712. The other controller is a Xylogics Model 732 Disk Controller that couples two floppy disk drives and one QIC02 tape drive to VMEbus1 systems.

1.1 USING THIS MANUAL

This manual describes the 732 floppy/QIC02 controller. Section 2 describes how to install and test the 732; Section 3 describes the 732 registers; Section 4 describes the IOPBs; and Section 5 describes the 732 commands. Section 6 describes error processing; Section 7 is a programming tutorial; Section 8 explains the 732's special functions; Section 9 describes the 732 theory of operation; and Section 10 includes maintenance aides.

1.1.1 Abbreviations

This manual uses the following mnemonics:

AIO	Add New IOPB
AIOP	AIO Pending
AIOR	AIO Response Time
AM	Address Modifier
ASR	Automatic Seek Retry
AUD	Auto-update
BHT	Black Hole Transfer
CHEN	Chain Enable
CMPL	Command Complete
CRIO	Clear Remove IOPB
CRBS	Clear Register Busy
CTYP	Controller Type
DFLT	Drive Fault
DMA	Direct Memory Access
DRDY	Drive Ready

- - - - - -

1. VMEbus is a trademark of the VMEbus International Trade Association.

1.1.1 Abbreviations (continued)

EDT	Enable DMA Timeout
ERRS	Error Summary
FERR	Fatal Error
FIFO	First In/First Out Disk Data Buffer
H	Notation For Numerical Values Expressed in Hexadecimal
ICS	IOPB Checksum
IEC	Interrupt At End Of Chain
IOPB	Input/Output Parameter Block
MBS	Megabytes Per Second
MMA	Maintenance Mode Active
MM	Maintenance Mode
NPRM	Non-privileged Register Mode
OVS	Overlap Seek Enable
PNUM	Prom Number
PRIO	Priority IOPB
PSEL	Priority Select
RBS	Register Busy Semaphore
REGCEL	Register Read/Write and Interrupt
RIO	Remove IOPB
RMM	Register Maintenance Mode
ROR	Release On Request
SGM	Scatter/Gather Mode
SKER	Seek Error
TDT	Throttle Dead Time
THRO	Throttle
TMOD	Transfer Mode
VMEDMA	Direct Memory Access Controller Chip
WRPT	Write-protect

1.2 DESIGN RELIABILITY

Xylogics implements the following features to minimize the likelihood of product failure:

- Design for worst case voltage and temperature.
- Extensive evaluation testing.
- Low parts count through extensive use of custom LSI.
- Buffer parity for continuous error checking.

1.2 DESIGN RELIABILITY (continued)

- o Low-stress design on all components.
- o All components burned-in.
- o One card; resides in backplane or expansion chassis.
- o Controller is power-cycled under thermal stress during test.

1.3 PHYSICAL

PACKAGING -- The 732 completely resides on part one printed circuit board.

DIMENSIONS -- The DCE is a Honeywell special size board.

CONNECTORS -- All connectors are on the edge of the board facing out.

1.4 ENVIRONMENTAL

The 732 environmental requirements are 0 - 55xC, with a maximum relative humidity of 90% (without condensation). Air flow across the board must maintain a maximum temperature differential of 7xC to prevent hot spots.

1.5 ELECTRICAL

POWER -- The 732 uses ? amperes at +5 volts DC (VDC).

TOLERANCE -- Voltage must be within plus or minus five percent (4.75 to 5.25).

GROUNDING -- Common earth ground must be established between the disk drives and the CPU chassis, backplane, and expansion cabinets.

1.6 SYSTEM RELATED SPECIFICATIONS

DATA BUFFERING -- The 732 has a FIFO buffer that is 8k-bytes long and incorporates parity error detection. Data can be put into one end of the FIFO and simultaneously removed at the other end; there are no delays associated with filling and emptying the buffer.

FORMAT -- The 732 Format command formats a specified number of tracks. Standard interleaving is 1:1.

STATUS LEDs -- The 732 implements two status LEDs. L1 (BSY) indicates the controller is active; L2 (ERR) indicates the on-board diagnostics did not complete successfully, or a fatal error occurred.

SCATTER/GATHER -- The 732 supports Scatter/Gather on Read and Write commands. The controller can gather data from various memory locations and transfer it to the buffer for use in a Write command; it can scatter the data out from the disk drive to the appropriate memory locations with a Read command. To execute a scatter/gather, software issues a normal Read or Write command along with a DMA list that contains a memory address and the number of words to transfer to/from that location. The smallest granularity of scatter/gather is a 16-bit word.

MULTIPLE IOPBS -- The 732 can store up to 15 IOPB's for processing, but only one will be active at any instant in time. Both the DMA data transfer and the drive activity must be completed before the next IOPB is started.

ERROR DETECTION -- The 732 supports a 16-bit data CRC on the floppy drives. The QIC02 tape drive formatter has its own error correction system.

BLACK HOLE TRANSFERS -- The 732 may transfer all the DMA data into the same bus address without incrementing the address at each DMA.

SOFTWARE SUPPORT -- Sample software driver supplied for use in UNIX2 (5.0) based systems (source included).

1.7 DISK DRIVE RELATED SPECIFICATIONS

PHYSICAL DRIVE INTERFACE -- The 732 supports the high density floppy disk drive.

INTERFACE DATA RATE AND STANDARD INTERLEAVE FACTOR -- The 732 supports a maximum disk data rate of 500 kilo bit per second (MBS). The 732 supports this data rate at a 1:1 interleave factor.

NUMBER OF DISK DRIVES -- The 732 supports up to two floppy disk drives.

DISK SECTOR FORMAT -- The 732 sector format includes a header field separated from a data field by a splice area.

HEADER FORMAT -- The header format is the industry standard 5.25" floppy format as generated by the WD37C65 controller chip.

CABLING -- The 732 uses standard 34 pin flat ribbon cable. The maximum length of each cable is 3 meters (9.8 feet).

1.8 QIC02 TAPE DRIVE RALATED SPECIFICATIONS

TAPE INTERFACE -- QIC02 formatted interface

TAPE DATA TRANSFER RATE -- The 732 supports tape drive data trasfer rates of up to 1 Megabyte per second. Typical QIC02 tape drives average data rates are 85 Kilo bytes per second.

NUMBER OF TAPE DRIVES -- Only one tape drive is supported. It is addressed as unit 4 and selects drive number 1.

INSTRUCTION SET -- The QIC02 standard instruction set is supported.

HARDWARE INTERFACE -- The standard QIC02 interface will be used except that all the receivers that are normally terminated with 150 to +5 will be terminated with 330 to +5 volts and 470 to 0 volts.

- - - - - -

2. UNIX is a trademark of AT&T.

1.9 VMEbus RELATED SPECIFICATIONS

VME COMPLIANCE NUMBER -- IEEE P1014/D1.0.

TRANSFER MODE -- Direct Memory Access (DMA).

DMA THROTTLE CONTROL -- Each time the 732 becomes bus master, it executes DMA transfers to or from the buffer up to the max throttle limit or the number of bytes/spaces available in the buffer.

DMA DATA TRANSFER RATE -- The 732 transfers data at a rate of up to 10 MBS; this rate requires Longword mode transfers and system memory that responds within 200 nanoseconds.

DMA DEAD TIME -- The 732 supports a programmable throttle dead time between throttle bursts. This prevents the 732 from taking over the bus and allows time for other DMA devices to access the bus.

DATA TRANSFER LIMIT -- Data transfer length, from 1 to 65,535 sectors with a single IOPB.

BUS COMPATIBILITY -- The 732 is compatible with the standard VMEbus.

ADDRESSING CAPABILITY -- The 732 supports Master A32, and Slave A16, as per the VMEbus Specification Manual. As a slave, the 732 responds to Address Modifiers 29H and 2DH.

DATA WIDTH -- The 732 supports D16 and D32 as per the VMEbus Specification Manual. The 732 transfers one byte, one word, or a byte and a word, until the transfer aligns with a word or longword boundary.

RELEASE ON REQUEST -- The 732 releases the bus at the request of other peripheral devices.

RELEASE WHEN DONE -- The 732 releases the bus after each bus access.

BUS REQUEST LEVELS -- The 732 supports four bus request levels.

EARLY RELEASE OF BUS BUSY/ -- The 732 does not support early release of Bus Busy/.

1.9 VMEbus RELATED SPECIFICATIONS (Continued)

INTERRUPT PRIORITY -- Software programmable interrupt level and vector.

CONTROLLER I/O PARAMETER BLOCK (IOPB) LENGTH -- 30 bytes.

CONTROLLER REGISTERS -- Seven 8-bit I/O Registers; byte or word addressable. Only eight bits respond during word access.

DIAGNOSTIC SUPPORT -- Comprehensive set of stand-alone diagnostics written in 'C' are available.

1.10 SOFTWARE RELATED SPECIFICATIONS

SOFTWARE INTERFACE -- The 732 supports a high level software interface that allows host software to use the same method to add IOPBs to a chain while the controller is busy or while it is free.

1.10.1 Software Interface

The software interface includes seven byte-wide registers. Four of these bytes comprise the VME Address Register, the fifth byte is the Address Modifier Register, and the sixth byte is the Control and Status Register (CSR). The CSR includes two bits that are very important to IOPB processing: Add IOPB (AIO) and Remove IOPB (RIO). The last byte is the Fatal Error Register; the 732 returns the fatal error codes in this register.

The IOPB is a block of command and status information; it includes the disk address, the bus address, and the type of operation to be performed. The software driver sets up the IOPB in user memory, sends the IOPB address to the VME Address Registers, and sets AIO. After the 732 receives the IOPB address it resets AIO. The 732 then performs the IOPB function and, upon completion or error, updates the IOPB status and sets RIO. The VME Address Registers point to the complete IOPB; the software driver reads the address and then resets RIO.

Software may add IOPBs to the queue, providing AIO is reset, by writing the IOPB address to the address registers and setting AIO (regardless of the 732's busy status).

1.11 PROGRAMMABLE FEATURES

- Software Controlled Interrupt or Polled Operations.
- Software Programmable DMA Parameters.
- Software Programmable Drive Size Parameters (Including Sector Size).
- Software Programmable Sector Interleaving -- Standard 1:1.
- Software Controlled Register Response.
- Software Controlled Transfer Retry/Correction.
- Software Programmable Hard or Soft Sector Mode.

1.12 COMPONENT SPECIFICATION

All components used on the 732 will be approved by Honeywell.

No 74S or 74AS parts will be used except for the 74AS286's FIFO parity generator/checker.

SECTION 2: INSTALLING AND TESTING THE 732

2.0 GENERAL

This secion describes how to unpack, configure, install, and test your 732 controller.

2.1 UNPACKING AND INSPECTION

2.1.1 Inspect the Shipping Carton

Inspect the carton for possible shipping damage. If you determine there is damage, do not unpack the unit. Notify Xylogics and the freight carrier immediately.

If no damage is visible, carefully unpack the 732. Save the carton and other packing material for possible later use.

2.1.2 Contents

The 732 is a single printed circuit board. Optional items include a manual and/or software on a floppy diskette, or + inch magnetic tape.

If any items are missing or damaged, please contact Xylogics at one of the following telephone numbers.

United States:	(617) 272-8140
United Kingdom (Milton Keynes):	0044-908-569444

2.1.3 Handling Precautions

Observing proper handling precautions minimizes the risk of damaging the 732 with electrostatic discharge. When transporting the 732, use an antistatic bag, antistatic bin, or the original shipping carton and packing material. Personnel handling the 732 should observe proper grounding methods including, but not limited to, wrist bands, heel straps, and antistatic mats.

2.1.4 Inspect the 732

Inspect the 732 for socketed parts that may have loosened during shipment. Assure that all parts are firmly seated in their sockets. If any parts must be reinserted, observe proper orientation.

2.2CONFIGURING THE 732

You can configure the 732 with several jumper options. The following subsections describe these options.

FIGURE 2-1. 732 - COMPONENT LOCATION

2.2.1 Base Address Selection

Jumper block JA controls the base address. Table 2-1 shows how to set the jumpers for commonly used base addresses. Inserting a jumper makes the 732 respond to a 0 on that address line; removing a jumper makes the 732 respond to a 1. Connect the jumper between similar pin numbers on each block. (The 732 uses bits 1 through 3 to determine which register is being accessed.) The 732 is an A16 Slave, and responds to address modifier 02DH, and optionally 029H.

* These two pins are test points, not address jumpers

FIGURE 2-2. BASE ADDRESS JUMPER BLOCK

Screen Label -->	F	E	D	C	B	A	9	8	7	6	5	4
Address:												
EEA0*	O	O	O	I	O	O	O	I	O	I	O	I
EE40	O	O	O	I	O	O	O	I	I	O	I	I
0800	I	I	I	I	O	I	I	I	I	I	I	I
0100	I	I	I	I	I	I	I	O	I	I	I	I

O = Out; I = In;

* Standard Factory Configurantion

TABLE 2-1. BASE ADDRESS SELECTION

2.2.2 Bus Request and Bus Grant Lines

The 732 uses the Bus Request and Bus Grant lines to become bus master. In VMEbus arbitration, there are four Bus Request/Grant levels: 0 through 3. The 732 drives one Bus Request line according to the jumper scheme you choose. The arbiter drives the four Bus Grant In lines: BG0IN* through BG3IN*. If the 732 receives a Bus Grant, and is not requesting the bus, it passes the grant by driving the appropriate Bus Grant Out line: BG0OUT* through BG3OUT*.

2.2.2 Bus Request and Bus Grant Lines (continued)

Select a request level by jumpering one Bus Request (BR0* through BR3*), one Bus Grant In, and one Bus Grant Out line to match the selected request level. Jumper the remaining Bus Grant In/Out lines so that the incoming signal passes through the board (i.e., jumper BGxIN* to BGxOUT*, where x represents the remaining grant levels).

For example, Figure 2-3 shows the jumpering scheme for level 0 (Figure 2-3A shows the jumper blocks as they actually appear on the board; 2-3B is labeled for this example): jumper JB1 to JB5; then jumper JC1 to JC5, and JD1 to JD5. Jumper the remaining Grant levels from JC6 to JD2, JC7 to JD3, and JC8 to JD4. Factory configuration: Bus Request Level 3.

NOTE

Some VME processors only support Bus Request Level 3.

FIGURE 2-3. JUMPERING BUS REQUEST AND BUS GRANT LEVELS

2.2.3 Parallel Arbitration

If you are using the 732 in parallel arbitration, and the Bus Grant Out lines must be isolated from the next slot's Bus Grant In Lines, remove all jumpers between JC 5-8 and JD 1-4 (See Figure 2-3B).

2.3SELF TEST DISABLE

When jumper JE 3-4 is installed, the 732 does not execute the Self Test on power-up.

2.4 PROMS AND PALS

LOCATION	PART NUMBER	TYPE

TABLE 2-2. PROM / PAL PART NUMBER AND LOCATION

2.5 LIGHT EMITTING DIODES

The 732 has two light emitting diodes (LEDs). L1 (BSY) is the Busy LED (it is located closest to the printed circuit board). L2 (ERR) is the Error LED (it straddles L1).

2.6 BOARD LABELS / REVISION CONTROL

All Xylogics controllers use various revision control labels. This information is important when discussing configuration issues with us. Please familiarize yourself with your board revision levels before contacting us.

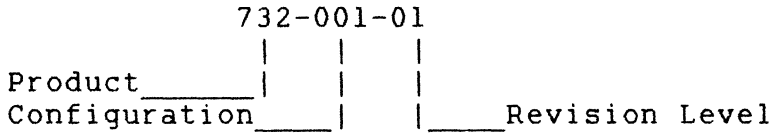


FIGURE 2-4. SAMPLE PART NUMBER

2.7 PREPARING THE COMPUTER SYSTEM FOR INSTALLATION

The backplane of your system must provide a VMEbus slot for the 732. The slot must be capable of handling a bus master, and the power source must handle the power consumption of the entire system, including the 732.

2.7.1 Backplane Jumpers

Remove any jumpers that short, or cause the Interrupt Acknowledge (IACK IN/OUT) and DMA Grants (BG 0-3 IN/OUT) to bypass the slot in which you are installing the 732.

2.7.2 Card Cage Slot

The card cage must have a slot at the proper DMA priority available for the 732. The 732 uses DMA to transfer data and IOPBs. Placement of the 732 in the DMA priority chain may be critical. The amount of bus bandwidth it uses will be high at times; this may affect other boards in the system. Likewise, other boards may not allow enough time for the 732 to DMA enough data to keep up with the disk; consider this when choosing a slot. If the 732 does not get a high enough priority, then its DMA falls behind what the disk requires, and it has to wait until the next revolution before continuing the transfer. If the 732 priority is high, it gets enough DMA time, but other boards having insufficient buffers may starve from lack of DMA time. The priorities must be balanced for your system to work properly.

2.7.3 Power Considerations

The 732 affects the power consumption of the entire computer system. The 732 uses +5 volts (4.75 to 5.25 volts) at ?? amps. Be sure the power supplies can handle the entire power load. Readjust the voltages AFTER plugging in the 732. A power supply that is just adequate may cause intermittent and unusual problems due to noise generated by occasionally going into overcurrent protection.

2.8 PREPARING THE DRIVES FOR INSTALLATION

Follow the manufacturer's instructions for unpacking and inspecting the drives.

2.9 INSTALL AND CABLE THE 732

2.9.1 Install the 732

Place the 732 into the computer card cage; make sure it is firmly seated. Be careful not to dislodge any socketed ICs. Situate the disk drive and connect it to its power source.

2.9.2 Cable the Subsystem

2.9.2.1 Connect the Floppy Cable.

Install the floppy cable, observing "pin 1" markings on both ends. This cable connects to the 34-pin connector on the 732, and to the cable connector on the drive. Use the "in" connector on the drive if there are two 34-pin connectors marked "in" and "out". The other connector should have a terminator, or the terminator should be built into the drive. (Only cable one disk drive for the initial system check. You can connect additional disk drives later.)

2.9.2.2 Connect The Tape Cable

Install a tape cable (50-pin cable) from the tape cable port on the 732 to the appropriate connector on the tape drive.

2.9.2.3 Mechanical Restraint

Make sure the floppy and tape cables are mechanically restrained at both ends to prevent them from accidentally disconnecting. Using "pull tabs" on the cables greatly reduces connector damage.

2.9.2.4 Disk Drive Grounds

Install a ground braid wire between the ground terminal on the disk drive(s) and the computer system ground.

2.10 INITIAL TESTS

This section relies upon your familiarity with your computer system's monitor and dignostics.

2.10.1 Power-up and Self Test

The 732 initiates a self test upon power-up. The Error LED (L2) lights for a moment, and then goes off. If L2 remains on, and the Fatal Error Register indicates an IRAM Checksum error, then you need to load good parameters into the IRAM. Otherwise, if L2 remains on, the board is not functioning properly (the Fatal Error Register may indicate the nature of the problem). Contact Xylogics for further assistance.

NOTE

Check the power supply voltage to ensure it is within limits (4.75 to 5.25 volts).

2.10.2 Drive Ready

Spin the drive up and wait for it to become ready. Issue a Read Drive Parameters IOPB. The Drive Status byte indicates the drive status at execution time. If DRDY is not set, recheck the drive cable connections and try again. If you are still unable to get the proper status, check the +5V supply on the bus. If the problem persists, check the disk drive for functionality with an off-line tester.

2.11 DIAGNOSTICS

When you run your diagnostics:

- Format the disk with either a diagnostic or format program.
- Run a full pass of your diagnostic (or determine that the system is working properly.
- Cable and test any additional drives.

2.12 CABLING MULTIPLE FLOPPY DRIVES

If you are using multiple drives, make sure the cables are properly connected; observe the "pin 1" markings on both the cables and the drives.

2.12.1 Terminator

Remove the terminator from the drive currently connected to the controller. Install the terminator in the last drive in the chain.

2.12.2 Unit Select - Floppy

If you are daisy-chaining drives, assign each drive a unique Unit Select number. The 732 accesses drives with Unit Numbers 1 and 2 for the floppy.

2.12.3 Unit Select - Tape

The 732 only accesses drive 1.

SECTION 3: THE 732 REGISTERS

3.0 GENERAL

The 732 programming interface is based on the use of seven, one-byte long, I/O registers. The bus address jumpers define the base address of the register set. Table 3-1 lists the registers along with the address offset from the base address. The 732 responds to either bytes or 16-bit words; when it responds to words, only 8 bits are valid.

The registers have one function when read, and another when written. The following subsections detail their definitions.

REGISTER	OFFSET
IOPB ADDRESS BYTE 0 (Least Significant Byte)	1
IOPB ADDRESS BYTE 1	3
IOPB ADDRESS BYTE 2	5
IOPB ADDRESS BYTE 3 (Most Significant Byte)	7
IOPB ADDRESS MODIFIER	9
CONTROL AND STATUS REGISTER	B
FATAL ERROR REGISTER	D

TABLE 3-1. REGISTER OFFSETS

3.1 IOPB ADDRESS REGISTERS

The first four registers define the 32-bit address of an IOPB or IOPB chain. When these registers are written, the 732 interprets it as the address of the IOPB or IOPB chain to be executed. When read, and Remove IOPB (RIO) is set, the registers point to the IOPB or IOPB chain completed by the 732.

The protocol for reading and writing this address register is defined by the use of the Add IOPB (AIO) and Remove IOPB (RIO) bits in the Control and Status Register (See Section 3.3).

3.2 IOPB ADDRESS MODIFIER REGISTER

This register defines the IOPB address modifier. (Address modifiers are used for many purposes, such as memory mapping, privilege levels, and addressing range. Please consult the VMEbus

3.2 IOPB ADDRESS MODIFIER REGISTER (continued)

Specification Manual for more information on using address modifiers.) This register also specifies whether an IOPB has priority over the current set of IOPBs in the 732 command queue. Section 3.3 defines the protocol for reading and writing this register.

PRIORITY IOPB REGISTER

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

RESERVED
ADDRESS MODIFIER

BIT	MNEMONIC	DESCRIPTION
7-6		RESERVED.
5-0	AM	ADDRESS MODIFIER - Most systems use the standard AM code of 3D. See the VMEbus Specification Manual.

3.3 CONTROL AND STATUS REGISTER

When written, this register provides the host with control of the 732 operation; when read, it provides the host with 732 status information. Section 3.3.1 defines the bits in this register when written; Section 3.3.2 defines the bits when read.

3.3.1 Control Register (Write)

CONTROL REGISTER (Write)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |

RESERVED
ENABLE MAINTENANCE MODE
RESERVED
CONTROLLER RESET
ADD IOPB
CLEAR RIO
CLEAR RBS

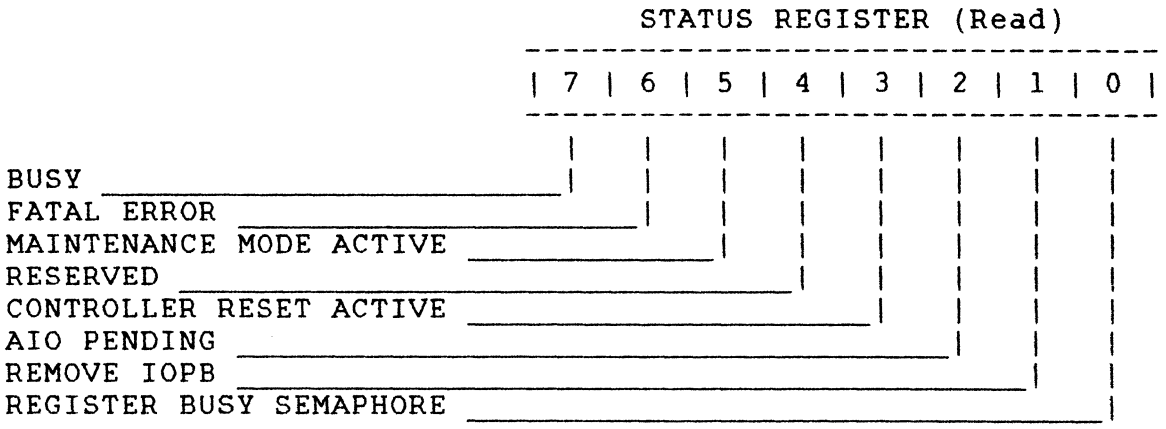
3.3.1 Control Register (Write) (continued)

BIT	MNEMONIC	DESCRIPTION
7-6		RESERVED.
5	MM	ENABLE MAINTENANCE MODE - Setting MM and AIO places the 732 in Maintenance mode. This mode supports a different Register protocol and is used as a diagnostic tool. Section 8 outlines the Maintenance mode.
4		RESERVED.
3	CRST	CONTROLLER RESET - This bit signals the 732 microprocessor to perform a "soft" reset; it deselects (releases dual port) all the drives, stops the DMA and Disk Sequencers (potentially during sector transfers), and cancels any IOPBs in the queue. When the Controller Reset completes, the 732 resets the CSR to zero. CRST does not initiate a Power-up Self Test. NOTE A Controller Reset takes up to one second to complete.
2	AIO	ADD IOPB - The host sets AIO to indicate that the 732 should execute the IOPB (chain) at the address pointed to by the IOPB Address and Address Modifier Registers. Essentially, AIO commands the 732 to begin executing a new IOPB (chain). As soon as the host asserts this bit, the 732 asserts the AIO Pending (AIOP) bit in the Status Register; this indicates that the 732 has received the AIO,signal but has not yet processed the address of the new chain. AIOP is negated in the Status Register after the 732 internally stores the new (chain) address. The 732 can store up to 47 IOPB addresses in this manner. Reasserting AIO if AIOP is asserted in the Status Register violates the Register protocol.

3.3.1. Control Register (Write) (continued)

BIT	MNEMONIC	DESCRIPTION
1	CRIO	CLEAR RIO - The host sets CRIO to clear RIO in the Status Register. Typically, the host sets CRIO after it reads the address of a completed IOPB chain from the IOPB Address and Modifier Regiaters. Clearing RIO enables the 732 to update the IOPB Address and Modifier Registers with the address and address modifier of a newly completed IOPB chain. Clearing RIO if it is not asserted in the Status Register violates the Register protocol.
0	CRBS	CLEAR RBS - The host sets the Clear Register Busy (CRBS) bit to clear the RBS bit in the Status Register. Clearing RBS effectively releases the registers for use by another host (See Section 8.8.2). (CRBS is only relevant in a multiprocessor environment.)

3.3.2 Status Register (Read)



BIT	MNEMONIC	DESCRIPTION
7	BUSY	BUSY - The 732 is executing IOPBs. The 732 sets BUSY when it clears AIOP to acknowledge the first IOPB address; it clears BUSY after completing all the IOPBs with no new ones pending (within 500 microseconds of the host clearing RIO on the last (IOPB). This bit is redefined when the 732 is in Maintenance mode (see Section 8.6).

3.3.2 Status Register (Read) (continued)

BIT	MNEMONIC	DESCRIPTION
6	FERR	FATAL ERROR - The 732 detected a fatal hardware error. A Controller Reset clears this bit. The 732 asserts FERR under the following conditions: (1) Maintenance Mode Test Failure; (2) Power-up Self Test Failure; (3) IOPB Checksum Miscompare; (4) IOPB DMA Fatal; (5) IOPB Address Alignment Error; (6) Firmware Error; (7) Illegal Maintenance Mode Test Number (8) ACFAIL Asserted.
5	MMA	MAINTENANCE MODE ACTIVE - When set, the 732 is in Maintenance mode (See Section 8.6).
4		RESERVED.
3	RSTA	CONTROLLER RESET ACTIVE - The host set Controller Reset in the Control Register and the 732 is currently resetting itself.
2	AIOP	AIO PENDING - When set, AIO has been set in the Control Register, but the 732 has not acknowledged its receipt. When clear, AIO may be set again. After the host reads the address and modifier, it must clear RIO by writing Clear RIO (CRIO) in the Control Register.
1	RIO	REMOVE IOPB - The 732 sets RIO after completing an IOPB, or a chain of IOPBs, and placing the address in the IOPB Address and Address Modifier Registers.
0	RBS	REGISTER BUSY SEMAPHORE - RBS provides a means of allowing multiple hosts to share access to the 732 registers without simultaneous access (See Section 8.8.2). (RBS is only relevant in a multiprocessor environment.)

3.4 FATAL ERROR REGISTER

If a fatal error occurs, the 732 returns the appropriate Completion Code in this register. Table 3-2 lists the fatal error codes; Section 6.5 describes them.

CODE	DESCRIPTION
E0	IRAM Checksum Failure
E1	IRAM Self Test Failure
E2	EPROM Checksum Failure
E3	Maintenance Test 3 Failure (Floppy Chip)
E4	Maintenance Test 4 Failure (QIC-02 Interface)
E5	Maintenance Test 5 Failure (VMEDMA Registers)
E6	Maintenance Test 6 Failure (REGCEL Chip)
E7	Maintenance Test 7 Failure (Buffer Parity)
E8	Maintenance Test 8 Failure (Disk FIFO)
E9-EF	Reserved
E0	IOPB Checksum Miscompare
F1	IOPB DMA Fatal
F2	IOPB Address Alignment Error
F3	Firmware Error
F5	Illegal Maintenance Mode Test Number

TABLE 3-2. FATAL ERROR CODES

SECTION 4: IOPB DESCRIPTION

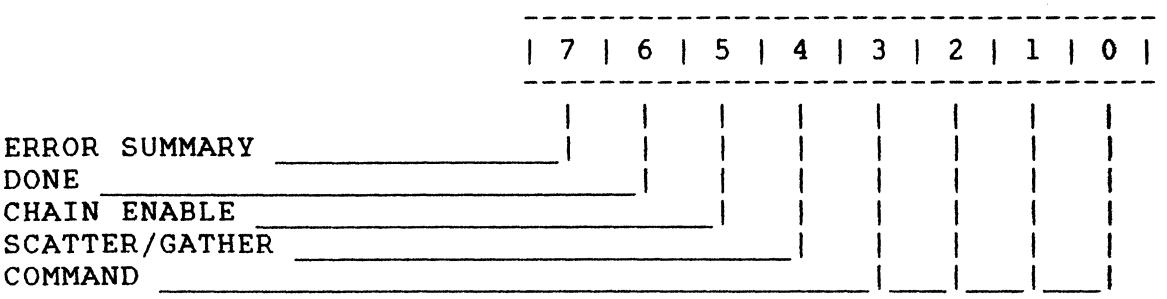
4.0 GENERAL

The Input/Output Parameter Block (IOPB) passes messages between the 732 and host software: software passes the type of transfer, disk address, data address, and count to the 732; the 732 returns the transfer status and possibly the ending addresses upon command completion. This section begins with the standard IOPB for most data transfer commands and follows with variations of the IOPB.

4.1 STANDARD IOPB

The 732 uses the standard floppy IOPB for data transfer commands and some general purpose commands, when a floppy unit number is selected . It uses the standard tape IOPB when the tape unit number is selected

4.1.1 IOPB Byte 0 (Command)



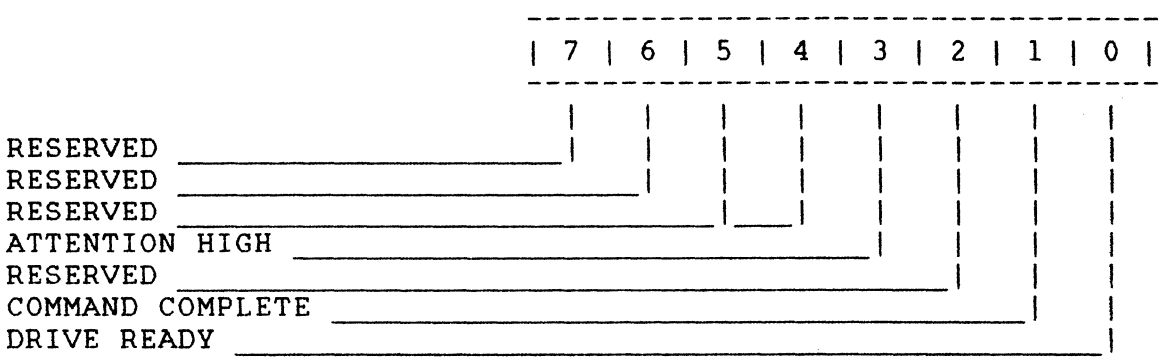
BIT	MNEMONIC	DESCRIPTION
7	ERRS	ERROR SUMMARY - ERRS is only valid if DONE is set. When set, a hard or soft error occurred during IOPB processing. When clear, the 732 successfully completed the IOPB.
6	DONE	DONE - When set, the IOPB is complete; if chained, software may remove the IOPB from the chain. Software must relink the chain in the same order as before, and cannot move IOPBs in memory.
5	CHEN	CHAIN ENABLE - When set, the Next IOPB Address Modifier and Next IOPB Address point to the next chained IOPB. When clear, this IOPB is not chained to another IOPB. If CHEN and IEC are set, the 732 returns the whole chain with one RIO; if CHEN is set and IEC is clear, the 732 returns one IOPB at a time.
4	SGM	SCATTER/GATHER MODE - When set, the IOPB is either a scatter (read) or a gather (write) transfer; a linked list describes the number of 16-bit words and to what address the 732 transfers each section of the data. The link address modifier and the link address specify the link list location. When clear, this IOPB specifies the data transfer address; the data is transferred to/from contiguous memory. SGM is only valid for standard Reads and Writes.
3-0	COMM	COMMAND - See Table 4-2.

4.1.2 IOPB Byte 1 (Status Byte 1)

After the 732 executes the IOPB, it sets DONE and posts a Completion Code in this byte. (Completion Codes are only valid if DONE is set.) A code of 0x indicates a successful completion; any other value indicates an error occurred (See Section 6).

4.1.3.1 IOPB Byte 2 (Status Byte 2) for FLOPPY DISK

IOPB Byte 2 is the Disk Status byte; it is only valid if DONE is set.



BIT	MNEMONIC	DESCRIPTION
7		RESERVED.
6		RESERVED.
5-4		RESERVED.
3	ATTN	ATTENTION HIGH - The 732 sets ATTN when the drive reports an error.
2		RESERVED.
1	CMPL	COMMAND COMPLETE - The 732 sets CMPL when the currently selected drive is ready to accept a command.
0	DRDY	DRIVE READY - The 732 sets DRDY when the last drive selected is ready.

4.1.3.2 IOPB Byte 2 (QIC02 status 0) for TAPE

BIT	MNEMONIC	DESCRIPTION
7	STO	SET - if any other bit in byte 0 is set. If bit 7 is set Exception may be set.
6	CNI	CARTRIDGE NOT IN - Exception set if cartridge removed, and (1) drive selected by select drive with Lock Cartridge command; (2) motion command is issued; or (3) tape moved previously from BOT.
5	DFF	DEVICE FAULT FLAG - No longer unselected drive (USL) bit. DFF sets Exception. Must be followed by a Read Status sequence to clear Exception. (Read Extended Status III contains information in Byte 25 to determine cause of fault.) DFF set when formatter detects 540 condition which prohibits further command execution. 1. No tape motion (jammed cartridge). 2. Failure to recognize or exit area between BOT/load point, or early warning/EOT. 3. No tach pulses. 4. Failure to complete command function in specified internal time. For example: not completing rewind once formatter initiates command. DFF indicates an unrecoverable 540 or cartridge error to user.
4	WRP	WRITE PROTECTED CARTRIDGE - Set if cartridge write protect mechanism on safe. Remains set if any Write or Erase command is issued when cartridge is write protected.
3	EOM	END OF MEDIA - Set when early warning hole detected on last track in write mode. Remains set while tape is at logical end of media. Not reset by Read Status. When set, Exception is set.

4.1.3.2. IOPB Byte 2 (QIC02 Status 0) for TAPE (continued)

BIT	MNEMONIC	DESCRIPTION
2	UDE	UNRECOVERABLE DATA ERROR - Set for unrecoverable data error during read or write operation. If set, Exception is set. Reset by Read Status.
1	BNL	BAD BLOCK NOT LOCATED - Set to indicate drive not able to locate correct block on tape. If set, Exception is set. When set with (UDE), drive transfers filler data block or data from a different block to keep correct total block count. BNL reset by Read Status.
0	FMD	FILEMARK DETECTED - Set when filemark block is read. Exception set and FMD reset by Read Status.

4.1.4.1. IOPB Byte 3 (Status Byte 3) for FLOPPY DISK

IOPB Byte 3 is reserved. It reflects the 732's internal status and may be a non-zero value.

4.1.4.2. IOPB Byte 3 (QIC02 Status 1) for TAPE

BIT	MNEMONIC	DESCRIPTION
7	STI	SET - If any other bit in Byte 1 is set. If set, Exception may be set.
6	ILL	ILLEGAL COMMAND - Exceptions and Bit 6 set under these conditions: 1. On line not asserted when read or write type command attempted or in process. 2. Non-implemented command is issued. 3. Non-read type command issued without proper termination of read sequence. 4. Non-write type command issued without proper termination of write sequence. ILL reset by Read Status.
5	NDT	NO DATA DETECTED - Set when drive determines no data is recorded on one tape. If set, Exception is set. NTD reset by Read Status.

4.1.4.2. IOPB Byte 3 (QIC02 Status 1) for TAPE

BIT	MNEMONIC	DESCRIPTION
4	MBD	MARGINAL BLOCK DETECTED - Set at detection of marginal data block. Enhanced track offset read recovery uses MBD to alert host if q4-mil or q8-mil offset required to read recorded cartridge. Exception set only if Exception and filemark read status are indicated. A set MBD indicates track position offset, when filemark was read. This status indicates to host a marginally recorded cartridge. Host may determine to write append tape, or recover data and rewrite cartridge.
3	BOM	BEGINNING OF MEDIUM - Bit set when drive is logically at BOT, Track 0. If set, Exception is set. Bit not reset by Read Status, but reset when tape moved away from logical BOT.
2	BPE	BUS PARITY ERROR - Bit set when drive detects odd parity error on bus during data transfer to drive. If set, Exception is set. Odd parity is an odd number of active bits on bus. Parity is enabled by W8, W9, W10 jumper configuration on 540 formatter. Only data checked for parity.
1	ERM	END OF RECORDED MEDIA - Bit set when drive detects end of recorded media, or following a Seek End of Data command. If set, Exception is set.
0	POR	POWER ON/RESET OCCURRED SET - Bit set following power on to drive or a reset from host. If set, Exception is set. Bit reset by a Read Status.

4.1.5 IOPB Byte 4 (Subfunction)

IOPB Byte 4 is the Subfunction byte. Subfunction Codes follow a convention that indicates whether the code is generic to all VME controllers, generic to a group of controllers (i.e., disk, tape, etc.), or specific to a particular controller (See Table 4-1).

4.1.5 IOPB Byte 4 (Subfunction) (continued)

The 732 combines standard Command Codes with Subfunction Codes to execute commands. The IOPB Command Code and Subfunction Code fields define the required operation. Table 4-2 lists the 732 Command and Subfunction Codes.

SUBFUNCTION CODES	CLASS
00-1F	Generic To All
20-3F	Generic Tape
40-5F	772-Specific
60-7F	Reserved
80-9F	Generic Disk
A0-AF	751-Specific
B0-BF	712-Specific
C0-FF	Reserved

TABLE 4-1. SUBFUNCTION CODE CLASSES

CODE	COMMAND	SUB FN	DESCRIPTION
0	NOP	00	No Operation
1	WRITE	00	Normal Write
2	READ	00	Normal Read
3	SEEK (FLOPPY)	00	Report Current Address
		01	Seek and Report Current Address
3	SEEK (TAPE)	~~21~~	~~Space record reverse~~
		40	File Mark Search Forward
		42	Mult. File Mark Search Forward
4	DRIVE RESET (FLOPPY)	00	Drive Reset (With RTZ)
4	DRIVE RESET (TAPE)	00	Drive Reset
		21	Rewind
		22	Offline
		24	Retension

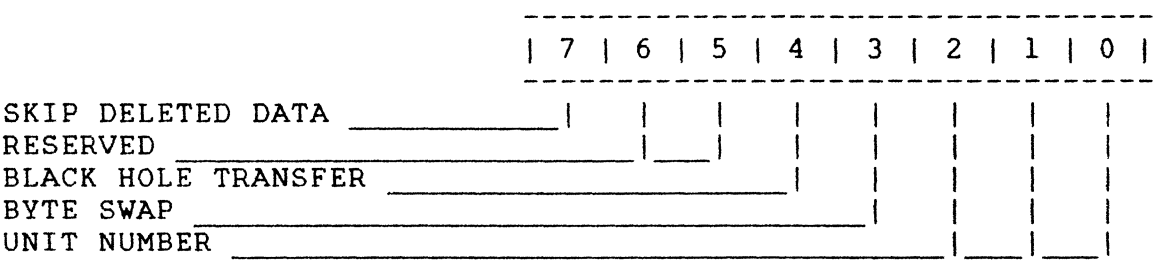
TABLE 4-2. 732 COMMAND/SUBFUNCTION CODES

4.1.5. IOPB Byte 4 (Subfunction) (continued)

CODE	COMMAND SUB FN	DESCRIPTION	
5	WRITE PARAMETERS	00	Write Controller Prmtrs.
		80	Write Drive Parameters
6	READ PARAMETERS	00	Read Controller Prmtrs.
		40	Read QIC02 Status
		80	Read Floppy Drive Parameters
7	EXTENDED WRITE (FLOPPY)	81	Write Track Format
7	EXTENDED WRITE (TAPE)	20	Write File Mark
		21	Erase
8	EXTENDED READ	81	Verify Data - not implemented
		83	Read Full Track
9	DIAGNOSTICS	00	Self Test
A-F	RESERVED		

TABLE 4-2. 732 COMMAND/SUBFUNCTION CODES (continued)

4.1.6 IOPB Byte 5 (Unit)



BIT	MNEMONIC	DESCRIPTION
7	SK	SKIP DATA FIELD - When set in Read Command. If deleted data mark detected, and go on to next sector. When clear in read command, if (floppy only)a DDM is detected, data is transferred fr that sector, and command stops with an error.

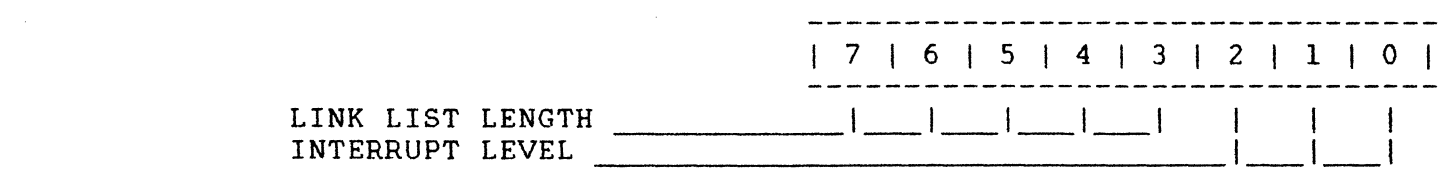
4.1.6 IOPB Byte 5 (Unit) (continued)

BIT	MNEMONIC	DESCRIPTION
7-5		RESERVED.
4	BHT	BLACK HOLE TRANSFER - When set, the 712 does not increment the bus address during a data transfer; IOPB transfers occur normally. When clear, the 732 does increment the bus address.
3	BS	BYTE SWAP - Each word transfer
2-0	UNIT	UNIT NUMBER - This value specifies the Unit Number of the attached drive to which the transfer is directed.

UNIT SELECTION

LOGICAL UNIT	BIT	BIT	BIT	DRIVE
0	0	0	0	Floppy Unit 1
1	0	0	1	Floppy Unit 2
2,3	0	1	X	Reserved
4	1	0	0	Tape Unit 1
5	1	0	1	Reserved
6,7	1	1	X	Reserved

4.1.7 IOPB Byte 6 (Interrupt Level)



BIT	MNEMONIC	DESCRIPTION
7-3	LLL	LINK LIST LENGTH - Bits 7-3 specify the length, in elements, of a linked list for Scatter/Gather commands. Each element refers to an 8-byte block in the linked list. See Table 8-2.

4.1.7 IOPB Byte 6 (Interrupt Level) (continued)

BIT	MNEMONIC	DESCRIPTION
2-0	INTL	INTERRUPT LEVEL - The 732 uses these bits as the VMEbus hardware interrupt level when it completes the IOPB. The 732 will not interrupt if bits 0 through 2 are clear.

NOTE

Depending on the command, Bytes 6 through 13 have different definitions (See Sections 4.2 through 4.4).

4.1.8 IOPB Byte 7 (Interrupt Vector)

IOPB Byte 7 determines the interrupt vector that the 732 uses upon command completion. This byte is not valid if the interrupt level is zero.

4.1.9 IOPB Bytes 8 and 9 (count)

Byte 8 is the Count High; Byte 9 is the Count Low. The bytes specify the number of disk sectors or tape blocks to be transferred in a data transfer IOPB.

The disk format command uses this count to specify the number of tracks to format.

The tape Multiple File Mark Search command uses this count to specify the number of contiguous file marks to be found.

If an error occurs, these bytes indicate the residual count, i.e. the number of sectors or blocks not transferred. A count of 0 is illegal, and the IOPB returns a completion code 013H.

4.1.10 IOPB Bytes A-D

These Bytes are used differently for floppy disk data transfer commands, tape data transfer commands and parameter commands. See section 4.2 and 4.3 for usage in parameter commands.

4.1.11 IOPB Bytes A-D for Floppy Disk

4.1.11.1 IOPB Byte A

This byte is reserved for floppy data transfer commands.

4.1.11.2 IOPB Byte B (Cylinder)

This byte specifies the starting cylinder address for a data transfer.

4.1.11.3 IOPB Byte C (Head)

This byte specifies the starting head address for a data transfer. Only heads 0 and 1 are valid.

4.1.11.4 IOPB Byte D (Sector)

This byte specifies the starting head address for a data transfer. The controller adds 1 to the logical sector number before addressing the drive, unless the SZV bit is set.

4.1.12 IOPB Bytes A-D for QIC02 Tape

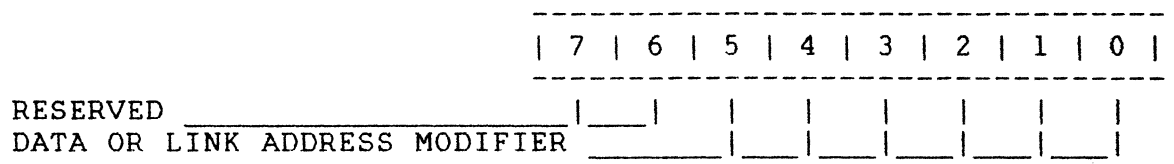
4.1.12.1 IOPB Bytes A and B

These bytes contain the Re-write count for a write operation if a QIC02 error occurred, or if the drive status is read.

4.1.12.2 IOPB Bytes C and D

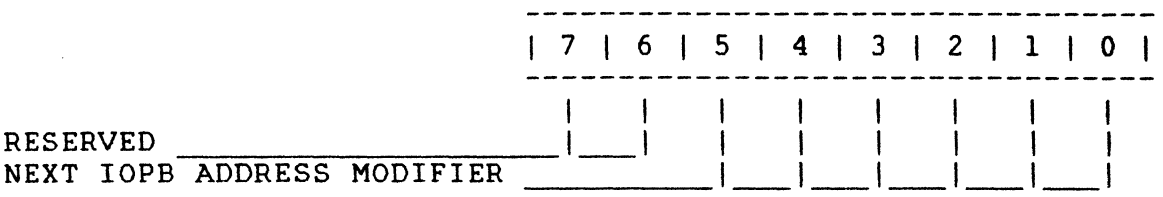
These bytes contain the Re-read count for a read operation if a QIC02 error occurred, or if the drive status is read.

4.1.13 IOPB Byte E (Data or Link Address Modifier)



BIT	DESCRIPTION
7-6	RESERVED.
5-0	DATA OR LINK ADDRESS MODIFIER - If SGM is set, bits 5 through 0 specify the Link List Address Modifier; if SGM is clear, this field specifies the Data Address Modifier. The 732 uses these modifiers to complete the address.

4.1.14 IOPB Byte F (Next IOPB Address Modifier)



BIT	MNEMONIC	DESCRIPTION
7-6		RESERVED.
5-0		NEXT IOPB ADDRESS MODIFIER - The Next IOPB address Modifier, along with the Next IOPB Address, point to the next IOPB in the chain.

4.1.15 IOPB Bytes 10 through 13 (DMA Data Address)

IOPB Byte 10 is DMA Data Address High; Byte 13 is DMA Data Address Low. These bytes comprise the data or link list address pointers. The 732 uses these bytes with the data or link list address modifier to point to the data or linked list address. If SGM is set, this address points to the linked list; if SGM is clear, this address points to the data address.

4.1.16 IOPB Bytes 14 through 17 (Next IOPB Address)

IOPB Byte 14 is Next IOPB Address High; Byte 17 is Next IOPB Address Low. These bytes comprise the Next IOPB Address pointers. The 732 uses these bytes with the Next IOPB Address modifier to point to the next IOPB in the chain (if CHEN is set in Byte 0).

4.1.17 IOPB Bytes 18 and 19 (IOPB Checksum)

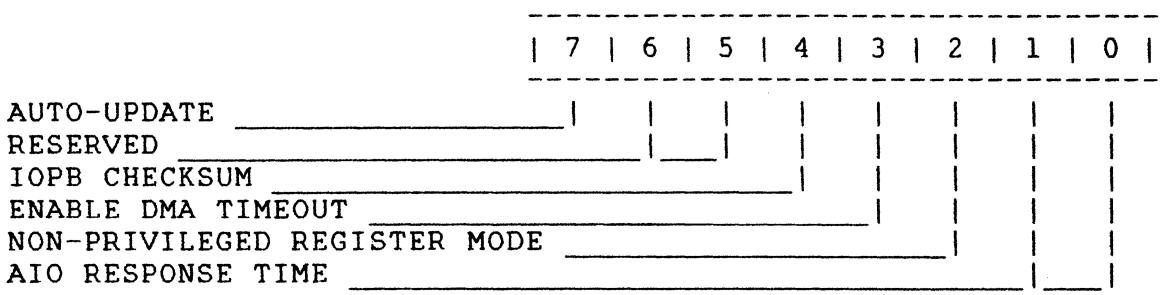
Byte 18 is IOPB Checksum High; Byte 19 is IOPB Checksum Low. The 732 calculates the checksum by adding the IOPB bytes. See Section 8.15.

4.2 CONTROLLER PARAMETERS IOPB

This IOPB sets and reads various controller parameters. The 732 uses the standard IOPB, but redefines bits in Bytes 8, 9, A, B, C, and E.

CONTROLLER PARAMETERS

4.2.1IOPB Byte 8 (Controller Parameters A)



BIT	MNEMONIC	DESCRIPTION
7	AUD	AUTO-UPDATE - When set, the 732 updates the IOPB to the transfer's correct ending parameters; it updates the disk address, the sector count, and the data address after completing the transfer or detecting an error. When clear, the 732 only updates the IOPB if an error occurs. The values are then set up so that host software can tell the 732 to continue (the values should point to the sector in error, the correct remaining sector count, and proper data address).

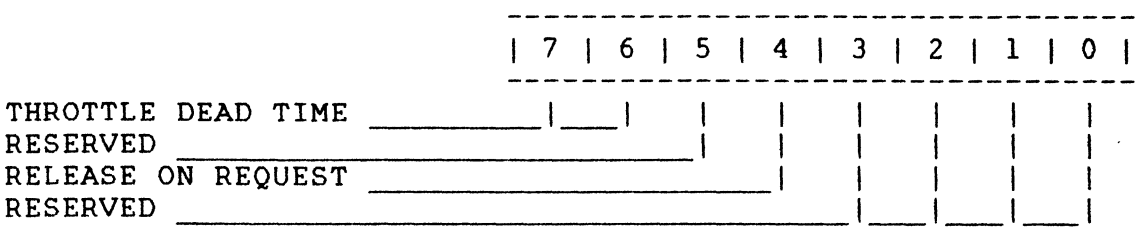
4.2.1 IOPB Byte 8 (Controller Parameters A) (Continued)

BIT	MNEMONIC	DESCRIPTION
6-5		RESERVED
4	ICS	IOPB CHECKSUM - When set, the 732 reads the IOPB, compares the checksum it generated during the read with the checksum the software driver appended to the IOPB. The 732 also updates the Checksum bytes in any IOPB if AUD is set. Clearing ICS disables this feature. See Section 8.15.
		NOTE Since this feature adds 50 microseconds to each transfer, it affects the 732's performance.
3	EDT	ENABLE DMA TIMEOUT - When set, the 732 enables a DMA bus error timer. When clear, the 732 relies on the VMEbus transfer timer.
2	NPRM	NON-PRIVILEGED REGISTER MODE - When set, the 732 responds to address modifiers 2DH and 29H. When clear, the 732 only responds to 2DH. (See the VMEbus Specification Manual for more information on address modifiers.)
1-0	AIOR	AIO RESPONSE TIME - These bits respond to the four values that indicate the maximum AIO response time. This is the time from setting AIO to the time the 732 clears it. The shorter the response time, the greater the 732 overhead.

VALUE	TIME
00	100 us (Factory Default)
01	75 us
02	62 us
03	50 us

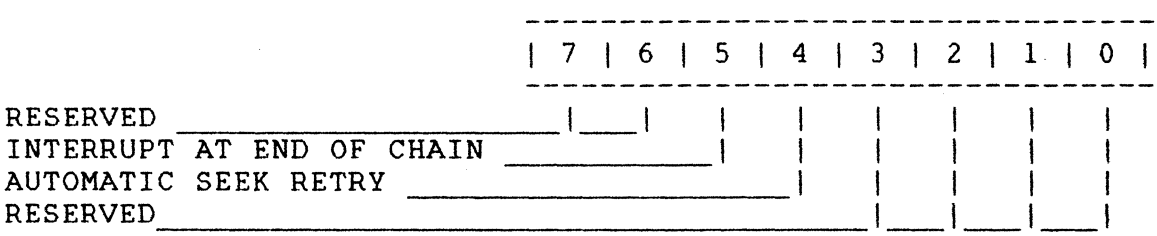
TABLE 4-3. AIO RESPONSE TIMES

4.2.2 IOPB Byte 9 (Controller Parameters B)



BIT	MNEMONIC	DESCRIPTION
7-6	TDT	THROTTLE DEAD TIME - TDT selects one of four minimum time periods that determines the time the 732 remains off the bus between throttle bursts (See Section 8.12).
5		RESERVED.
4	ROR	RELEASE ON REQUEST - When set, the 732 releases the bus at the request of other bus masters; otherwise, it continues with the next throttle burst. The 732 monitors the bus request lines and releases bus busy only if another bus request is pending. It completes its specified throttle burst before releasing the bus due to a pending request. When clear, the 732 releases the bus at the end of each throttle burst and rearbitrates if more data transfers are pending.
3-0		RESERVED.

4.2.3 IOPB Byte A (Controller Parameters C)



BIT	MNEMONIC	DESCRIPTION
7-6		RESERVED

4.2.3. IOPB Byte A (Controller Parameters C) (Continued)

BIT	MNEMONIC	DESCRIPTION
5	IEC	INTERRUPT AT END OF CHAIN - When set, the 732 returns all IOPB chains with one RIO and one interrupt; it does not relink or unlink IOPBs. The RIO address of a completed chain is the address of the first IOPB in the chain. The 732 also uses the interrupt level and vector of the first IOPB in the chain. Clearing IEC disables this feature. (Do not set or clear IEC while the 732 is processing an IOPB chain.)
4	ASR	AUTOMATIC SEEK RETRY - When set, the 732 resets the drive, seeks to the commanded cylinder and retries the transfer on Seek and Header Error/Cylinder errors.
3-0		RESERVED.

4.2.4 IOPB Byte B (Controller Parameters D)

Bits 0 through 7 are the Throttle (THRO) bits. The throttle is the maximum number of transfers allowed each time the 732 becomes bus master. The throttle value determines the maximum DMA burst length for both data and IOPB DMA transfers. Each bit position represents a binary weight, allowing a throttle from 1 to 256.

VALUE	WEIGHT
0	256
1	1
2	2
3	3
:	:
255	255

TABLE 4-4. THROTTLE VALUES

4.2.5 IOPB Byte C (Release Level)

The 732 returns its release level on a Read Controller Parameters command.

4.2.6 IOPB Byte E (Controller Type)

IOPB Byte E is the Controller Type byte. Xylogics assigns each VME controller a unique controller type code.

CONTROLLER	CODE (H)	
712	12	(ESDI Disk Controller)
732	32	(Floppy & QIC02 Controller)
751	51	(SMD/SMD-E Controller)
772	72	(Pertec Tape Controller)

TABLE 4-5. CONTROLLER TYPE CODES

4.2.7 IOPB Bytes 10 and 11 (EPROM Part Number)

The 732 returns a portion of the EPROM part number on a Read Controller Parameters command. The 4 nibbles in these 2 bytes refer to the part number's last 4 digits. For example, if the part number is 180-002-151, Byte 10 holds 21H and Byte 11 holds 51H.

4.2.8 IOPB Byte 12 (Revision)

This byte contains the revision level of the EPROM plugged into the board (0=Unreleased Prototype, 1=A, 2=B, etc.).

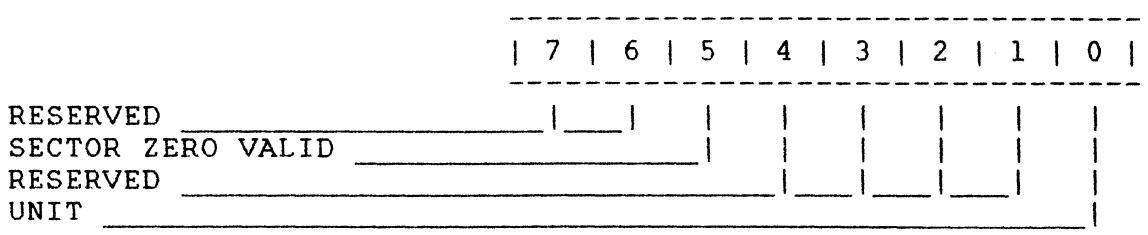
4.2.9 IOPB Byte 13 (Subrevision)

This byte contains the subrevision level of the EPROM plugged into the board. Any value other than zero indicates that the microcode is an unreleased version available for testing purposes (0=Released, 1=1, 2=2, etc.).

4.3 DRIVE PARAMETERS IOPB (Floppy only)

DRIVE PARAMETERS

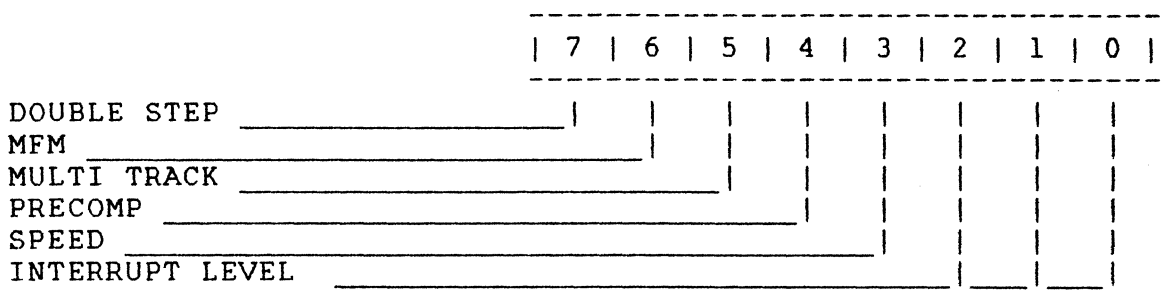
4.3.1. IOPB Byte 5 (Unit)



4.3.1 IOPB Byte 5 (Unit) (continued)

BIT	MNEMONIC	DESCRIPTION
7-6		RESERVED.
5	SZV	SECTOR ZERO VALID - If set, the first sector on the media sector is 0. If clear, the first sector on the media is sector 1 (normal mode).
4-1		RESERVED.
0	UNIT	UNIT - If set command applies to Drive 2, if clear command applies to Drive 1.

4.3.1 IOPB Byte 6 (Drive Parameters)



BIT	MNEMONIC	DESCRIPTION
7	DSTP	DOUBLE STEP - When the seek distance from the present cylinder to the required cylinder is doubled, and that number of step pulses is issued to the drive. It allows a 48TPI disk to be read on a 96TPI drive.
6	MFM	When set the media is read and written in MFM mode. When clear, FM encoding is used.
5	MT	MULTI TRACK - When set, if the max head is 1, i.e. the drive has 2 heads, large transfers spiral from the last sector on head 0 to the first sector on head 1 through to the last sector on head 1 to the first sector of the next cylinder of head 0.

4.3.1 IOPB Byte 6 (Drive Parameters) (continued)

BIT	MNEMONIC	DESCRIPTION
4	PCMP	PRECOMPENSATION - When set all MFM data is precompensated by q125ns, regardless of cylinder. When clear, MFM data written to cylinders less than 28 are precompensated by 187ns.
3	SPD	SPEED - When set dual spindle speed drives are run at 300 revs/minute. When clear, the drive runs at 360 revs/minute.
2-0	INTL	INTERRUPT LEVEL

4.3.2 IOPB Byte 8 (Head Load Time)

Byte 8 is the Head Load Time in milliseconds, values 2-254ms. Recommended value for the TOSHIBA ND-08DE is 36ms.

4.3.3 IOPB Bytes 9 and A (Gap 3 length)

4.3.3.1IOPB Byte 9 specifies the length of GAP 3 used in Read and Write operations.

4.3.3.2IOPB Byte A specifies the length of GAP 3 used in Format operations.

4.3.3.3Recommended values of GAP 3 for 5," drives (from NEC fPD765A manual).

MFM	MODULATION	SECTOR SIZE	N	R&W GPL	FORMAT GPL
0	FM	128	0	07	09
0	FM	256	1	10	19
0	FM	512	2	18	30
0	FM	1024	3	46	87
1	MFM	256	1	0A	0C
1	MFM	512	2	20	32
1	MFM	1024	3	2A	50
1	MFM	2048	4	80	F0

4.3.4 IOPB Byte B (Max Cylinder)

IOPB Byte B specifies the maximum cylinder number to be accessed on the selected drive. For an 80 cylinder drive the value will be 79 = 4F hex.

4.3.5.IOPB Byte C (Sector Size and Max Head)

Bits 7-4 specify the number of bytes per sector, they represent the variable 'n' recorded in the media header.

n	bytes per sector
0001	256
0010	512
0011	1024

Bits 3-1 are reserved and must be set to 0. Bit 0 if set, specifies a drive with two heads. If clear, the drive has only one head or only head 0 will be used on the media.

4.3.6. IOPB Byte D (Max Sector)

IOPB Byte D specifies the drive's maximum sector value. This value is 0 based. For example, on a drive with 8 sectors, the maximum sector is specified as 7. All starting sector numbers for data transfers are specified in the range 0-7. If the SZV bit is set, data transfer is started from the sector with that value on its header. If the SZV bit is clear, one is added to the sector number in the IOPB, and the data transfer is started from that sector, as there is no sector 0 on the media.

4.3.7. IOPB Byte E (Timing)

4.3.7.1. Head Unload Time

Bits 7-4 specify the time to delay before unloading the heads after a data transfer operation has completed. The value is increments of 16ms from 1=16ms to F=240ms. The user may select any value, short times give minimum head wear, but longer access time for a follow-on command which must start with a head load.

4.3.7.2. Step Rate

Bits 0-3 specify the 2's complement of the interval in milliseconds between stepping pulses sent to the drive. 0=16ms, F=1ms. Recommended value for the TOSHIBA ND-08DE drive is 3ms, value D hex.

SECTION 5: COMMANDS

5.0 GENERAL

Each disk command begins a new page. An IOPB diagram follows each command description. The diagrams are highlighted to indicate which bytes the 732 requires for command execution, and which bytes return after execution.

Each 732 IOPB is 26-bytes long. All commands use Bytes 0 through 19H. Reserving all 26 bytes maintains IOPB integrity.

5.0.1 Setting Up the Command

Each IOPB diagram indicates the bytes or fields that must be set for each operation. Certain parameters are essential; others are optional. All commands require the Command, Unit, and Interrupt Level fields to contain valid information. (This is also true for the Interrupt Vector field if the Interrupt Level is not zero.)

5.0.2 Completing the Command

After the 732 completes the command, it updates IOPB Bytes 0 through 3 with ERRS, DONE, a Completion Code, and an internal status. The 732 only updates the entire IOPB if Auto-update (AUD) is enabled, an error occurs, or if Read Parameters, Report Current Address or Read Status commands are executed. If AUD is set, and no errors occur, the 732 sets DONE, posts a Completion Code of zero in Byte 1, and disk drive status information in Byte 2; for any command that DMAs data to/from memory, the 732 updates the data address to point to the last address plus one of the transfer. See Table 5-1.

STATUS	ACTION
AUD Clear/No Error Occurs	732 updates Bytes 0-3 with ERRS, DONE, Completion Code, and internal status
AUD Set/No Error Occurs	732 updates entire IOPB
AUD Clear/Error Occurs	732 updates the entire IOPB
AUD Clear/A Read Params Report Current Address or Read Status Command is Executed	732 updates the entire IOPB

TABLE 5-1. 732 COMMAND COMPLETION

5.1 NO OPERATION

The No Operation (NOP) command is a diagnostic tool. The 732 reads the IOPB and marks it complete.

NOP

5.2 WRITE DATA

The 732, after reading and decoding the IOPB, positions the disk drive heads at the target cylinder; it then reads in the data from the host (indicated by the IOPB) and writes the data contiguously to the disk's sequential sectors.

Write Data has two IOPB formats: Normal and Scatter/Gather. A Normal IOPB specifies one contiguous block of host memory to write to the disk. A Gather Write IOPB specifies up to 32 different blocks of host memory to be placed in contiguous sectors on the disk (See Section 8.11).

If the BS bit is set in IOPB Byte 5, the high and low byte in each word are swapped.

WRITE DATATAPE

5.3 READ DATA

The 732, after reading and decoding the IOPB, positions the disk drive heads at the target cylinder, then reads the disk data indicated by the IOPB, and writes the data in host memory.

Read Data has two IOPB formats: Normal and Scatter/Gather. A Normal IOPB specifies one contiguous block of host memory that is used when placing the data from the disk. A Scatter Read IOPB specifies up to 32 different blocks of host memory where the disk data will be placed (See Section 8.11).

If the BS bit in IOPB byte 5 is set, the high and low byte in each word are swapped. See Section 4.1.6 for handling of deleted data dependant on the SK bit.

READ DATATAPE

5.4 REPORT CURRENT ADDRESS (Floppy only)

The 732 selects the disk drive, reads the first good header field, and returns the address to the host via the IOPB; it updates the IOPB regardless of AUD's status.

REPORT CURRENT ADDRESS

5.5 SEEK AND REPORT CURRENT ADDRESS (Floppy only)

The 732 issues a seek to the selected disk drive for the target cylinder. After the drive completes the seek, the 732 reads the first good header field it encounters and reports it to the host via the completed IOPB. The 732 updates the IOPB regardless of AUD's status.

SEEK AND REPORT CURRENT ADDRESS

5.6 POSITION (Tape only)

The position commands move tape forward and reverse over records or files, without transferring data. The 732 uses the count field to set up the number of file marks to search for.

POSITION

5.6.1. Position Subfunction Codes

CODE	DESCRIPTION
21	SPACE RECORD REVERSE - Spaces reverse one record. Each command places the tape head in the interrecord gap that separates records from one another (see figure 5-1). If the 732 detects a tape mark, tape motion ceases and the controller reports a File Mark Detected On Read error. The 732 aborts this command if it encounters a BOT marker (Reverse into BOT error).
40	FILE MARK SEARCH FORWARD - Searches forward the number of file marks specified in the count field. Each count placed the tape heads in the interrecord gap just after the file mark in questions. If this command is issued, and there are no file marks on the tape, the 772 searched until it detects EOT.
42	MULTIPLE FILE MARK SEARCH FORWARD - Searches forward for a specified number of consecutive file marks. This is especially useful for positioning the tape heads at the logical end of tape (usually indicated by several consecutive file marks). This command follows the same completion rules as File Mark Search Forward.

FIGURE 5-1. TYPICAL TAPE FORMAT

5.7 DRIVE RESET

5.7.1. Floppy Disk

The 732 issues commands to the disk drive to reset. First it issues a fault clear, and then a recalibrates (return to zero). The IOPB is complete when the recalibrate completes or times out on drives that are ready. The 732 does not wait for the recalibrate to complete on drives that are not ready.

5.7.2.Tape

The 732 issues a drive reset to the QIC02 interface.

DRIVE RESET TAPE

5.7.3 Drive Reset Subfunction Codes

CODE	DESCRIPTION
00	DRIVE RESET - Resets the tape drive by pulsing the reset line. Use this command if a tape runaway condition occurs. Always consider the tape position unknown following a Drive Reset.
21	REWIND - Moves the tape at high speed in the reverse direction until the BOT marker is detected. The tape drive should indicate it is at BOT or load point. The Rewind command completes immediately, although the tape is still rewinding. To verify the tape drive has completed rewinding, software must issue a Read Drive Status command. The following conditions should be true: BOT set, REW clear, and DRRDY set. (Caution: even if REW is clear, the drive may not be ready [DRRDY clear].)
22	OFFLINE - Removes the online signal from the QIC-02 interface, causing the drive to write a file mark if the previous command was a write, and to initialise a rewind command. All other commands leave the drive online.
24	RETENSION - Issues a command to the QIC02 drive to go to the end of tape and rewind to the beginning of medium.

5.8 WRITE CONTROLLER PARAMETERS

This command initializes the 732 with its operational parameters. No default parameters are assumed, but once loaded, the parameters remain stored in the 732 volatile memory until the controller is powered down. Section 4.2 defines how to change the parameters for individual applications; Section 6.6 explains the IRAM checksum.

WRITE CONTROLLER PARAMETERS

5.9 WRITE DRIVE PARAMETERS (Floppy only)

This command informs the 732 of the disk drive's physical characteristics. No default values are assumed, but once loaded, the parameters remain stored in the 732 volatile memory until the controller is powered down. See Section 4.3.

The disk address bytes OB-OD specify the maximum cylinder, head and sector on the media.

WRITE DRIVE PARAMETERS

5.10 READ CONTROLLER PARAMETERS

The 732 returns the current 732 operational parameters to the host via the IOPB; it verifies the IRAM checksum before completing the transfer regardless of AUD's status. See Section 4.2.

READ CONTROLLER PARAMETERS

5.11 READ DRIVE PARAMETERS (Floppy only)

The 732 returns the disk drive's physical characteristics to the host via the IOPB. The 732 verifies the IRAM checksum before completing the transfer; it updates the IOPB regardless of AUD's status. See Section 4.3.

READ DRIVE PARAMETERS

5.12 READ STATUS BYTES (Tape only)

The 732 reads the QIC02 drive interface's status bytes. It returns drive-specific status and diagnostic information in the IOPB. Consult your drive manufacturer's manual for the definitions for read status 0. This command is always sent to the drive by the controller if an exception occurs.

READ STATUS

5.11 WRITE TRACK FORMAT (Floppy only)

The Write Track Format command directs the 732 to format the drive, writing the header of all sectors with the appropriate sector ID. The data field contains zeros and a valid ECC. The Count bytes in this command refer to the number of tracks to be formatted. See Section 8.3.

WRITE TRACK FORMAT

5.12 EXTENDED WRITE (Tape only)

This command controls two separate Write functions: Write File Mark and Erase. These functions require different IOPB formats. Section 5.12.1. explains these functions and follows with IOPB diagrams.

5.12.1 Write File Marks

CODE	DESCRIPTION
20	WRITE FILE MARK - The 732 issues a Write File Mark command to the tape drive. File marks are special records that logically group data records on tape. Host software should write at least one file mark, preferably two, at the logical end of the tape. (Write File Mark uses the count field to specify the number of file marks to write.)

WRITE FILE MARK

5.12.2 Erase Tape

CODE	DESCRIPTION
21	This command erases the whole tape from BOM to EOT and rewinds to BOM.

ERASE

5.13 VERIFY DATA - NOT IMPLEMENTED

This command verifies the data on the disk. The 732 reads the data from the host and the disk simultaneously, and compares them on a bit-by-bit basis. The granularity of the mismatch reporting is one sector. The ending data address does not indicate where a mismatch error occurred.

VERIFY DATA

5.14 DIAGNOSTICS

The 732 executes the on-board self test diagnostics. Do not chain this IOPB to another IOPB. It cannot be used in conjunction with other IOPBs in the command queue.

DIAGNOSTICS

SECTION 6: ERROR PROCESSING

6.0 GENERAL

The Error Summary (ERRS) bit, Fatal Error (FERR) bit, and Completion Code represent the 732's status after executing a command. FERR indicates the transfer failed and the 732 requires a Controller Reset before continuing. ERRS only affects the specific IOPB and may be tested in lieu of checking the Completion Code; the 732 does not require a Controller Reset before continuing (see 6.3). The Completion Code informs software that the 732 successfully completed a command, failed to complete a command, or encountered and corrected a problem with one of several internal recovery procedures.

6.1 THE COMPLETION CODE

The 732 posts a Completion Code in IOPB Byte 1 (Status Byte 1); a Completion Code is only valid if DONE is set. Table 6-2 lists the Completion Codes (all codes not listed in the table are reserved). The following subsections describe these codes, along with any required corrective action.

6.1.1 Completion Code Convention

Completion Codes follow a convention that indicates the action required by either the software driver or manual intervention. The byte's upper nibble is the recovery code, and the lower nibble is the actual error code.

RECOVERY CODE	RECOVERY PROCEDURE
0	No Action / Status Only
1	Non-retryable Programming Error
3	Successfully Recovered Soft Error
4	Hard Error / Retry
6	Hard Error / Reset and Retry
7	Fatal Hardware Error
8	Miscellaneous Error
9	Requires Manual Intervention

TABLE 6-1. RECOVERY CODE

6.1.1 Completion Code Convention (continued)

ACTION	CODE (HEX)	DEVICE	DESCRIPTION
No Action/ Status Only	00		Successful Completion
Non-retryable Programming Errors	10	F	Illegal Cylinder Address
	11	F	Illegal Head Address
	12	F	Illegal Sector Address
	13	F,T	Count Zero
	14	F,T	Unimplemented 732 Command
	1C	F,T	Illegal Scatter/Gather Length
	1E	F,T	Next IOPB Alignment Error
	1F	F,T	Scatter/Gather Addr.Alignment
	21	F,T	Illegal Black Hole Address
Successfully Recovered Soft Errors	32	F	Auto Seek Retry Recovered
	33	F	Soft Retry Recovered
Hard Errors/ Retry	40	F	Hard Data CRC
	41	F	Header Not Found
	42	F,T	Drive Not Ready
	43	F,T	Operation Timeout
	44	F,T	VMEDMA Timeout
	45	T	QIC02 Parity Error
	46	F,T	FIFO Parity Error
	4A	F,T	Fatal VMEDMA Error
	4B	F,T	VMEbus Error
Hard Errors - Reset/Retry	60	F,T	Drive Faulted/Write Fault
	61	F	Header Error/Cylinder
	62	F	Header Error/Head
	63	T	Tape Exception
	64	F	Seek Error
Fatal Hard- ware Errors	70	F	Illegal Sector Size
	71	F,T	Firmware Failure
Miscellaneous Errors	81		IRAM Checksum Failure
	83	T	IOPB Aborted by Error
Requires Manuald Intervention	90	F,T	Write-protect Error

TABLE 6-2. SUMMARY OF COMPLETION CODES

6.1.2 Completion Code Descriptions

6.1.2.1 No Action / Status Only

Typically, the following Completion Codes require no action; the 732 returns the codes for status only.

CODE(H)	DESCRIPTION
00	SUCCESSFUL COMPLETION -- Not an error; indicates the command is complete and the IOPB may be removed from the queue.

6.1.2.2 Non-retryable Programming Errors

This group of errors usually occurs while debugging drivers; they should not occur in a normal operating system environment.

CODE (H)	DEV	DESCRIPTION
10	F	ILLEGAL CYLINDER ADDRESS -- Host software specified a cylinder address greater than the maximum cylinder number specified in the last Set Drive Parameters command for this drive. Correct the cylinder address, and retry the IOPB operation.
11	F	ILLEGAL HEAD ADDRESS -- Host software specified a head address greater than the maximum head address specified in the last Set Drive Parameters command for this drive.
12	F	ILLEGAL SECTOR ADDRESS -- Host software specified a sector address greater than the maximum sector number specified in the last Set Drive Parameters command for this drive.
13	F,T	COUNT ZERO -- Host software issued the 732 an IOPB that required a count, but the count was zero. Read, Write and Format commands require a valid count.
14	F,T	UNIMPLEMENTED CONTROLLER COMMAND -- This error occurs on all reserved 732 commands.

6.1.2.2 Non-retryable Programming Errors (continued)

CODE (H)	DEV	DESCRIPTION
1C	F,T	ILLEGAL SCATTER/GATHER LENGTH -- The linked list specified a number of words to transfer that does not agree with the amount of data contained in the requested number of sectors for transfer.
1E	F,T	NEXT IOPB ALIGNMENT ERROR -- The Next IOPB Address did not start on a 16-bit boundary; the 732 does not execute the NIOPB.
1F	F,T	SCATTER/GATHER ADDRESS ALIGNMENT ERROR -- A Scatter/Gather address started on a byte boundary.
21	F,T	ILLEGAL BLACK HOLE ADDRESS -- During a Black Hole Transfer, the data address started on a byte boundary.

6.1.2.3 Successfully Recovered Soft Errors

This group of errors is for status only. If some errors recur often, the operating system should try to map out the sectors involved. Allowing these errors to recur degrades performance.

CODE (H)	DEV	DESCRIPTION
32	F	AUTO SEEK RETRY RECOVERED -- This is a soft error. The 732 completed the transfer successfully but, during the transfer, it had to reset the drive to recover from an error.
33	F	SOFT RETRY RECOVERED -- The 732 encountered an error while executing this command. A retry was successful.

6.1.2.4 Hard Errors/Retry

These errors indicate the transfer failed; retry the operation. If several retries fail, manual intervention is required or the operating system may crash.

CODE (H)	DEV	DESCRIPTION
40	F	HARD DATA CRC ERROR -- The 732 detected a hard data CRC error in the data field during a Read command. Retry the previous Read operation.
41	F	HEADER NOT FOUND -- The 732 cannot find the requested sector. The controller searches for a match for at least one disk revolution plus five sectors to locate the header.
42	F,T	DRIVE NOT READY -- The selected drive is not ready, but not faulted; issue a Drive Reset. Causes include: o Drive not up-to-speed. o Drive hardware error. o Bad or improperly connected cable(s). o No drive of the specified Unit Number is connected to the 732.
43	F	OPERATION TIMEOUT -- The 732 did not complete the IOPB within a two second timeout period.
43	T	OPERATION TIMEOUT -- The 732 did not complete the IOPB within a timeout period that depends on the command: REWIND, ERASE, RETENSION, FILEMARK SEARCH 5 minutes READ, WRITE, WRITE FILEMARK 40 seconds OTHERS 2 seconds
44	F,T	VMEDMA TIMEOUT -- The DMA controller did not complete within its timeout. One reason could be that memory did not respond in time.

6.1.2.4 Hard Errors/Retry (continued)

CODE (H)	DEV	DESCRIPTION
45	T	QIC02 PARITY ERROR -- The QIC02 tape drive reported an exception with the parity error bit set.
46	F,T	FIFO PARITY -- The transfer failed; the 732 detected a FIFO parity error.
4A	F,T	FATAL VMEDMA ERROR -- The VMEDMA stopped for no apparent reason. The count nor the address overflowed, and there was no bus error.
4B	F,T	VMEBUS ERROR -- The VME BERR* signal was asserted while the 732 was bus master (See the VMEbus Specification Manual).

6.1.2.5Hard Errors - Reset/Retry

This group of errors indicate the transfer failed. Software should issue a Drive Reset command to the drive in use before retrying the operation.

CODE (H)	DEV	DESCRIPTION
60	F,T	DRIVE FAULTED / WRITE FAULT -- The selected drive is faulted. Issue a Drive Reset. If the fault persists, you must intervene.
61	F	HEADER ERROR/CYLINDER -- The cylinder address did not match during a sector search. Check the cylinder address and retry the operation.
62	F	HEADER ERROR/HEAD -- The head address did not match during a sector search.

6.1.2.5 Hard Errors - Reset/Retry (continued)

CODE (H)	DEV	DESCRIPTION
63	T	TAPE EXCEPTION -- The QIC02 tape driver completed a command with the Exception line active. This error is not reported if a file mark search command completes successfully, even though the exception line is active.
64	F	SEEK ERROR -- The disk drive reported a seek error.

6.1.2.6 Fatal Hardware Errors

These errors indicate the hardware failed. Manual intervention or a Controller Reset may be the only recovery approach.

CODE (H)	DEV	DESCRIPTION
70	F	ILLEGAL SECTOR SIZE -- The disk drive's sector size is not large enough to hold the header, data, and specified field lengths.
71	F,T	FIRMWARE FAILURE -- Flag settings or counter values are inconsistent with the firmware routines being executed. Document the conditions and call Xylogics.

6.1.2.7 Miscellaneous Errors

81		IRAM CHECKSUM FAILURE -- The calculated checksum from the IRAM and its stored value did not match during the Self Test or read parameters command. The parameters that are in error are not necessarily in the parameters read by this IOPB; they may be elsewhere in the IRAM. Check all the programmable parameters. Any write parameters command resets the checksum, and any subsequent read parameters will be error free. A soft bit in the IRAM, static, or probing the board with the power on can cause this error. See Section 6.6.
83	T	IOPB ABORTED BY ERROR -- This IOPB has been aborted because a previous error has left the tape drive in an unknown state. See Section 6.3.

6.1.2.8 Requires Manual Intervention

The write-protect error requires you to manually remove the write-protection.

CODE (H)	DEV	DESCRIPTION
90	F,T	WRITE-PROTECT ERROR -- A command that writes to the device (e.g., Write, Format, Write Track Headers) is issued, but the drive is write-protected.

6.2 SOFT ERROR COMPLETION CODES

The 732 updates the IOPB with the last error it encounters; it may overwrite previous soft errors with a new soft error status or a hard error status.

6.3 ABORT BY ERROR

If a hard error occurs on a tape IOPB, all further IOPB's are "aborted by error", and are not decoded, except for Read Drive Status and Drive Reset Commands.

6.4 FATAL ERROR CODES

If a fatal error occurs, the 732 sets FERR in the Status Register and posts the error code in the Fatal Error Register. (The following error codes appear only in the Fatal Error Register.) The only way to clear a fatal error is by issuing a Controller Reset (CRST).

CODE	DESCRIPTION
E0	IRAM CHECKSUM FAILURE -- The IRAM checksum did not match the expected checksum following bus initialization.
E1	IRAM SELF TEST FAILURE -- The 732 writes the IRAM with an incrementing data pattern then reads it with a decrementing pattern. An error indicates a bad IRAM.
E2	EPROM CHECKSUM FAILURE -- At power-up, the EPROM checksum did not match the IRAM checksum. Either the EPROM is degraded, or the IRAM changed during power-down.

6.4 FATAL ERROR CODES (Continued)

CODE	DESCRIPTION
E3	MAINTENANCE TEST 3 FAILURE -- The 732 writes and reads the registers in the Floppy Disk Chip.
E4	MAINTENANCE TEST 4 FAILURE -- The 732 tests the QIC02 interface.
E5	MAINTENANCE TEST 5 FAILURE -- The 732 writes the VMEDMA Registers and then reads them. An error indicates a bad VMEDMA.
E6	MAINTENANCE TEST 6 FAILURE -- There is a problem with the REGCEL chip.
E7	MAINTENANCE TEST 7 FAILURE -- The FIFO parity circuit failed its diagnostic.
E8	MAINTENANCE TEST 8 FAILURE -- The 732 fills the Disk FIFO with sequential data and then reads it. An error indicates a problem with the DSKCEL or FIFO.
F0	IOPB CHECKSUM MISCOMPARE -- The generated checksum did not match the appended checksum. This error can only occur while IOPB checksum feature is active. ICS is controlled via controller parameters. See Section 8.15.
F1	IOPB DMA FATAL -- The 732 did not complete the DMA within the prescribed timeout period. The memory could be defective or not present; the 732 may not have been able to become bus master.
F2	IOPB ADDRESS ALIGNMENT ERROR -- The IOPB address did not start on a 16-bit boundary. Change the address of the IOPB and retry.
F3	FIRMWARE ERROR -- Flag settings or counter values are inconsistent with the firmware routines being executed; the IOPB cannot DMA the appropriate error status. The 732's state is indeterminate; you must issue a Controller Reset.
F5	ILLEGAL MAINTENANCE MODE TEST NUMBER -- The command is invalid, or the Maintenance mode jumper is not in.

6.5 IRAM CHECKSUM

Each time the 732 executes a read parameters command, it compares a generated checksum with the stored checksum. This checksum encompasses the area that contains all the parameters, not just the ones being read. When this error occurs, the checksums did not match; rewrite or check all the parameters. Any write parameters command generates and stores a new checksum.

SECTION 7: A TUTORIAL IN PROGRAMMING THE 732

7.0 GENERAL

This section describes programming the 732 for basic use. This tutorial programming procedure begins with a single NOP IOPB and progresses to normal Read and Write commands. Each section builds on the previous section's information. (In the Sent/Returned portion of each sample IOPB, the x represents an indeterminate value that depends on the external conditions.)

7.1NO OPERATION (NOP)

The NOP command allows you to become familiar with the 732 programming interface.

FIGURE 7-1. SAMPLE NOP IOPB

7.1.1 Allocating Memory for an IOPB

First, allocate space in host memory to store the IOPB. This allocation is a function of the operating system or the program that is currently executing. Next, set up the IOPB to execute a simple NOP command.

7.1.2 Point the 732 to the IOPB

The IOPB is now in host memory. Point the 732 to the IOPB by loading the IOPB address and address modifier into the appropriate 732 registers. Make sure the address compensates for any memory mapping that may be done between virtual and physical addressing in your system. The 732 looks for the IOPB at the physical address to which the registers point.

7.1.3 Starting the Operation

The 732 now points to the IOPB in host memory. Writing the AIO bit in the CSR directs the 732 to process the IOPB.

7.1.4 732 Operation

At this point, the 732 performs the following functions:

1. Clears AIOP and sets BUSY.
2. Reads the IOPB from host memory.
3. Decodes the command.
4. Performs the operation (NOP).
5. Sets the DONE bit.
6. Updates the IOPB.
7. Puts the completed IOPB's address into the registers.
8. Sets RIO.
9. Clears BUSY.

7.1.5 Command Completion

Software has been polling RIO (since interrupts are not enabled [Interrupt Level = 0]). Software knows that the 732 sets RIO when it is done. Software should get the completed IOPB's address from the registers, and then clear RIO. This completes the NOP command.

NOTE

Do not poll the DONE bit in the IOPB. The 732 sets DONE while the rest of the IOPB is still updating.

7.1.6 Returned Values

DONE is set in the returned IOPB. If unit 0 or 1 is selected, Status Byte 2 reflects the status of the Disk Drive and Status Byte 3 reflects the 732's internal status. If unit 4 is selected Status Bytes 2 and 3 reflect the QIC02 Status Bytes 0 and 1.

NOTE

Status Byte 3 is proprietary to Xylogics and may change definition without notice.

7.2 READ CONTROLLER PARAMETERS

Next, implement a Read Parameters command with a Controller Parameters subfunction (See Section 4.2). This command returns several controller parameters in the returned IOPB.

FIGURE 7-2. SAMPLE READ CONTROLLER PARAMETERS IOPB

7.2.1 Execute the IOPB

Set up the IOPB in host memory; point the 732 to the IOPB. Set AIO and the 732 executes this IOPB.

7.2.2 732 Operation

The controller operation changes slightly from the example in Section 7.1.4.

The 732 performs the Read Controller Parameters operation instead of the NOP. The controller gets the parameters from its internal store, and puts them in the proper IOPB locations. The 732 fully updates the IOPB, including the returned values.

While reading the controller parameters, the 732 calculates a new internal RAM (IRAM) checksum and compares it to the previous value. The 732 returns the appropriate Completion Code if the values do not match.

7.2.3 The Returned IOPB

The values in the returned IOPB describe the last setting of the software-programmable parameters. Determine if each value works for your application. After making any necessary changes, write the parameters back to the 732.

Specific bytes have known values. The Controller Type byte contains a 12H; the PROM Part Number byte contains 21H and 51H. See Section 4.2 for more information.

7.3 WRITE CONTROLLER PARAMETERS

Next, write the controller parameters. Xylogics recommends reading the current parameters, modifying the ones in question, and then writing them back to the 732. This method allows you to change only those parameters that affect your system.

7.3 WRITE CONTROLLER PARAMETERS (continued)

FIGURE 7-3. SAMPLE WRITE CONTROLLER PARAMETERS IOPB

7.3.1 732 Operation

The 732 executes the IOPB slightly different than in Sections 7.1.4 and 7.2.2: it performs this function by taking the values of all programmable parameters out of the IOPB and setting the appropriate flags and variables in its internal code. It also calculates a new checksum in the IRAM and stores it for use in the next reading of any parameters.

7.4 READ/WRITE DRIVE PARAMETERS (Floppy only)

The Drive Parameters commands allow you to configure the 732 to your drive's size and parameters. Section 4.3 describes the size and configuration variables that may be modified with these commands. The operation is similar to controller parameters. A separate parameter is stored in the 732 for unit 0 and unit 1.

FIGURE 7-4. SAMPLE WRITE DRIVE PARAMETERS IOPB

7.4.1 732 Operation

On a Read or Write Drive Parameters command, the 732 performs an operation similar to that of controller parameters.

7.4.2 Execute the IOPB with Interrupts

To build on 732 functionality, enable interrupts for this example by specifying an interrupt level and vector.

7.4.3 732 Operation

The 732 performs the operation almost identically to the example in Sections 7.2.2 and 7.3.1, but with an additional step. After the 732 sets RIO, it performs an interrupt sequence.

7.4.4 Command Completion

Enabling interrupts modifies the command completion. Software does not poll RIO when it is set, but may be off doing something else (probably waiting for an interrupt). When the interrupt occurs, hardware and software execute an Interrupt Service Routine (ISR) and process the interrupt. Hardware resets the actual hardware interrupt when the ISR is called.

The ISR reads the address of the completed IOPB from the registers, and clears RIO. This completes the Read/Write Drive Parameters operation.

7.5 FORMAT A TRACK

Up to this point we have been initializing the 732. Initialization informs the 732 of the drive size, and parameters it requires before it can properly function. Now, let's format one track of the floppy disk drive. The 732 can only execute Read and Write commands on a formatted track. (Typically, formatting is done only once in the lifetime of the media.)

7.5 FORMAT A TRACK (continued)

FIGURE 7-5. SAMPLE WRITE TRACK FORMAT IOPB

7.5.1 732 Operation

The Format command is the first command in this tutorial that transfers data from the controller to the disk. The 732 operation for data transfer commands differs greatly from initialization commands.

7.5.1 732 Operation (continued)

To format a track:

1.The 732 still clears AIOP, sets BUSY, and reads the IOPB from memory. The next step occurs after the IOPB is in the 732.

2.The 732 decodes the function, and determines if a seek is required. All Write, Read, Write Extended, and Read Extended functions require the drive to seek to the commanded cylinder. Format is a Write Extended function; it requires the drive to seek. The 732 issues a seek to the drive by commanding the FDC chip to issue the required step pulses to the drive.

3.The 732 waits for the drive to complete the seek; the drive indicates it's done by returning command complete.

4.The 732 loads the data for the each sector header into the F.D. RAM, waits for index, and writes the new header on Sector 0; it writes the data field with zeros, and writes the data field CRC.

5.It then writes zeros for the inter-sector gap and writes the header and data field for the next sector.

6.The 732 repeats Step 5 for each sector on the track, until another index pulse occurs when the formatting stops. It is essential that software has set up the drive paramaters so that the correct number of complete sectors fit on the track.

7.The 732 updates the IOPB with the ending values, and completes the command.

7.6 VERIFY THE FORMAT

Now that the track is formatted, the headers may be verified either by reading or writing each sector on the track. The read operation will both verify the headers and the integrity of the zero data patterns written on the media. The write operation will verify the headers, but must be followed by a read command to verify the data, which can now be a non-zero test pattern.

Setup either a read or write command with a count equal to the number of sectors per track, and allocate a space in host memory for the data buffer. The buffer length must be the sector size times the number of sectors per track. Put the physical address of the start of the allocated data buffer into the IOPB data address bytes.

7.7 WRITE DATA

This subsection describes a Write operation, and the following subsection describes reading back the data. Allocate space in host memory for the buffer, and set up a data pattern in this buffer; an incrementing count in the buffer will suffice.

FIGURE 7-7. SAMPLE WRITE DATA IOPB

7.7.1 732 Operation

The 732 operation is similar to the previous examples; the differences are in DMAing data into the FIFO, and writing data to the disk.

The 732 starts the DMA from host memory to the FIFO immediately; it enables the FDC write command when the FIFO contains one full sector of data, and the drive is on cylinder.

The 732 compares and verifies the header: the FDC tests all the headers as they pass under the head, until it finds the sector designated for transfer. At the proper point in the sector, the 732 writes a new Sync byte, and then the data it read from memory. The 732, using the data to be written, generates and appends a CRC on the end of the sector.

7.7.2 Command Completion

The command is complete as soon as the FDC completes its operation. The 732 puts the ending values into the internal IOPB, and performs an appropriate update.

7.8 READ DATA

The 732 writes the data to the drive on Sector 0, Head 0, and Track 0. This subsection describes reading back the data and verifying it. You must allocate a data buffer for the 732 to write the data in memory. After allocation, it is a good idea to fill the buffer with a known pattern that differs from the expected data.

7.8.1 732 Operation

The 732 treats this command like the previous operations, except in the way it reads the data from the disk, and DMAs the data from the FIFO.

The 732 enables the FDC as soon as the drive is on-cylinder. After the controller finds the correct header, it transfers the data from the disk to the FIFO. As soon as the first word of data is available from the buffer, the DMA controller DMAs the data from the FIFO to host memory. The transfer is done when the DMA controller completes the DMA.

7.8.2 Command Completion

The 732 completes the command when the DMA to memory is complete.
The next subsection describes how to verify the data.

FIGURE 7-8. SAMPLE READ DATA IOPB

7.8.3 Verify Data

First, make sure the buffer was modified. If it was not modified, either an error occurred, or software specified the wrong buffer address. Next, compare the data written with the data read; they should match.

7.9 MULTIPLE SECTOR TRANSFERS

You can repeat the steps in Sections 7.8 and 7.9 using a larger sector count. The 732 crosses head and cylinder boundaries, as required, to complete the required number of sectors. Be sure to allocate enough buffer space for the increased sector count.

7.10 SUMMARY

This section was an exercise in testing the 732's functionality in your system. The steps are basically the same when the software driver controls the 732. (Operating systems always allocate the buffers.)

SECTION 8: 732 SPECIAL FUNCTIONS

8.0 GENERAL

This section describes how to implement the various 732 special functions. Each subsection descibes how minor functions implement a given major function.

8.1 MEDIA DEFECT HANDLING

8.1.1 Floppy Disks

There are no facilities in the 732 for media defect handling. Either the systems software must handle any media defects, or perfect floppy disk media should be used.

8.1.2 QIC02 Tapes

The 732 has no facilities for handling media defects, but the QIC02 will skip over bad media during write operations, to give the impression to the 732 that the media is perfect.

8.2 CHAINING AND MULTIPLE I/O REQUESTS

The 732 has two ways of speeding up multiple IOPB execution. One method allows the driver to chain IOPBs together, and then give the 732 a command-chain. The second method allows the driver to add IOPBs to the 732's queue by the same procedure as starting the first IOPB.

8.2.1 Chaining

Each IOPB has a Chain Enable (CHEN) bit and a Next IOPB pointer. IOPBs can be chained together by setting CHEN and having the Next IOPB pointer point to the next IOPB to be executed. Each IOPB in the chain points to the next, and, in order to stop the chain, CHEN is not set in the last IOPB.

NOTE

The Next IOPB Address is the physical address, not the virtual address.

8.2.2 Multiple I/O Requests

The following procedure allows you to add IOPBs to the 732 queue:

1.AIOP must be clear. If it is not clear, wait; it normally clears within 100 microseconds.

2.Write the five IOPB address registers to point to the beginning of the IOPB or IOPB chain.

3.Write the AIO bit.

8.2.3 732 Operation

The 732 treats IOPBs the same, regardless of how they were added to the queue.

8.3 FORMATTING THE FLOPPY DISK

This subsection describes formatting, including how to set the gaps and setting the disk size.

8.3.1 Specify Sector Data Size

The drive parameters IOPB specify the max cylinder, head and sector and the sector size for the unit specified in the IOPB. See section 4.3

8.3.2 Specify Sector Gap Size

Bytes 9 and A of the drive parameters IOPB specify the size of gap 3 for the Read/Write and Format commands respectively. Gap 3 for the format is the number of bytes between the CRC of a data block and the sync bytes of the next sector.

Gap 3 for the Read and Write commands is the number of bytes that the FDC ignores after a data block before starting the PLO to lock onto the preable of the next sector. It skips over the write splice so that the next header sync pattern is recognised correctly.

8.4 ERROR RECOVERY

The 732 may automatically retry operations that have errored. The write controller parameters command enables or disables the Autoseek retry bit.

8.4.1 Automatic Operation Retry

The 732 automatically retries an operation if the reason for the initial failure is a seek error. Setting ASR with a Write Controller Parameters command enables this option.

8.5 MAINTENANCE MODE

Firmware supports a non-IOPB driven Maintenance mode. It allows you to perform basic testing within the 732 by setting Control bits in the CSR and entering the desired test number and data through the address registers. This firmware also provides a window through which internal registers may be examined or modified.

8.5.1 Register Use in Maintenance Mode

The function code in the Test Number Register determines whether or not the 732 uses the Input Data Byte and Output Data Byte Registers (See Table 8-1). You should be familiar with the Control and Status Register before reading this section (See Section 3.3).

REGISTER	DESCRIPTION
1	Test Number or Function Code
3	Input Address Low
5	Input Address High
7	Input Data Byte (If Required)
9	Output Data Byte (If Required)
B	Control and Status Register
D	Fatal Error Register

TABLE 8-1. REGISTER USE IN MAINTENANCE MODE

8.5.2 Maintenance Mode Protocol

8.5.2.1 Executing a Maintenance Command or Entering the Maintenance Mode

First, set the Maintenance Mode (MM) and AIO bits. This forces entry into the maintenance kernel. The kernel initializes the CSR and Poll mask and sets the Remove IOPB (RIO) bit; then clear RIO.

The kernel expects the Input Address Low Registers to contain a maintenance test number or function code for execution. Data may be expected or may be returned (see register layout). BUSY and AIO are configured for polling. Setting BUSY selects the register image test; clearing BUSY returns control to the maintenance kernel.

AIO causes the maintenance firmware to read and decode the command string from the Input Address Registers. After successfully decoding the command string, the firmware echoes it (command, address, and data) to the Output Address Registers and clears AIO. This acknowledges receipt of and attempts to execute the requested command. After completing the requested command, the 732 updates the Output Address Registers with test-pertinent data and sets the RIO bit. The AIO/RIO protocol is identical to Normal mode. (RIO indicates the end of firmware involvement and valid contents in the Output Address Registers.)

Since each test and its expected results are different in nature, the Output Address Registers hold the test result information (address, data, etc.). In any case, the firmware sets RIO upon command completion; it sets the Fatal Error bit if a failure occurs or if host software issues an illegal command.

8.5.2.2 Exiting the Maintenance Mode

To exit the Maintenance mode, clear MM and RIO, and set AIO. This returns control to the Normal mode kernel. The 732 acknowledges by setting RIO.

8.5.2.3 Diagnostic Considerations

The Input/Output Address Register Verify is the first test the diagnostic should execute. Firmware flags the Power-up Test failures by setting the Fatal Error bit while leaving the Maintenance mode bit set. Firmware saves the Self Test error numbers internally until it verifies the Input and Output Registers.

8.5.2.4 Register Tests

You must request entry into the Maintenance mode to invoke the Register test. After the firmware acknowledges the request, you should set the BUSY bit. BUSY remains set during this test.

NOTE

You must enter the Maintenance mode as a separate step because the Normal mode firmware does not allow setting BUSY (defined as RMM when Maintenance mode is enabled).

Writing the Input Address Registers, followed by AIO, signals the firmware to copy the data to the Output Address Registers. Firmware sets RIO when it completes the copy. Host software should then clear RIO.

Clearing the BUSY bit exits this test and returns the 732 to Maintenance mode.

8.5.2.5 Test Variables

Some of the internal tests require the address and data to perform their particular function. On-board memory has space allocated for this data. These locations are loaded with default values for initial use. However, you may alter these variables through the Manual mode. (As the internal tests are defined, the protocol and results expected will be made available.)

8.6 MULTIPROCESSOR SUPPORT

The 732 has several options that make multiprocessor environments easier to support: the programmable interrupt vector, interrupt level, register address modifiers, and busy semaphore.

8.6.1 Interrupts

Each IOPB specifies the interrupt level and vector for that command. In a multiprocessor environment, each processor can have its own assigned interrupt level and vector.

8.6.2 Register Busy Semaphore

RBS allows multiple processors to share the registers without colliding. Hardware supports the RBS bit. The register access protocol involves reading the CSR. If RBS is clear, the host has control of the register, and retains control until it clears RBS in the Control Register. If the first read to the Status Register indicates that RBS is set, then another host has control of the register and this host must wait until RBS clears.

The 732 sets RBS immediately after a host reads the CSR. If a host attempts a read, and RBS is clear, then the 732 sets RBS; any successive reads by other hosts will "see" that RBS is set. When the host using the registers is done, it must clear RBS. Clearing RBS and setting AIO can occur in the same register write. Clearing RBS without having control of the registers violates the register protocol.

8.6.3 Address Modifiers

The address modifiers can be used to assign separate address space for each of the processors.

8.7 SOFTWARE CONTROL

The 732 has many parameters that can be modified by software control. The parameters can be set in bulk with three write parameters commands. The Write Drive Parameters command modifies the drive parameters. The Write Controller Parameters command modifies the controller parameters.

8.7.1 Modifying a Single Parameter

The best method for modifying a single parameter is to first do a read parameters of the associated parameter block, modify the single parameter, and then write the parameter block back to the controller.

8.7.2 Modifying a Group of Parameters

Use the same method as in Section 8.7.1, or set all the parameters in the specific IOPB and execute the appropriate write parameters command. The 732 sets all parameters to the new values contained in the IOPB.

8.7.3 Parameter Reference Point

After the 732 is working as intended, read the parameters and save the information for future use.

8.7.4 Setting Parameters at Boot Time

It is necessary to reload the parameters at each boot since the parameters are stored in a volatile RAM.

8.7.5 Validate Current Parameters

The parameters are all protected by a checksum, and any read parameters command performs a checksum test. Any read parameters terminates with an error if the generated parameter checksum is different than the stored checksum (See Section 6.6).

8.8 SCATTER/GATHER

The Scatter/Gather feature is used in conjunction with standard Read and Write commands. In a Scatter Read, the 732 transfers the data to up to 32 blocks of memory. Gather Writes gathers data from up to 32 blocks of memory and writes it to the disk. The size of each memory block must be an even byte count and less than 64K-bytes long. The blocks may be scattered throughout memory.

8.8.1 Scatter/Gather Link List

You can determine the length of the linked list by multiplying the number of elements in the list by eight (each element is eight-bytes long). All data addresses must be on word boundaries and the byte count must be even. For Read and Write operations, enter the number of elements in the linked list into IOPB Byte 6, bits 3 through 7. A zero in this field indicates the linked list has 32 elements. See Tables 8-2 and 8-3.

8.8.1 Scatter/Gather Link List (continued)

LINK NUMBER	BYTE	DESCRIPTION
1	00-01	Byte Count (Multiples of 2)
	02	Reserved
	03	Data Address Modifier
	04-07	Data Address (Word Boundaries Only)
2	08-09	Byte Count
	:	
	:	
n	xx	

TABLE 8-2. SCATTER/GATHER LINK LIST

LINK FIELD VALUE	DECIMAL EQUIVALENT
0	32
1	1
2	2
:	:
9	9
A	10
B	11
:	:
1E	30
1F	31

TABLE 8-3. LINK LIST FIELD VALUES

8.8.2 Setting Up a Scatter/Gather Transfer

The Data Address and Modifier bytes in the IOPB should now point to the start of the linked list. The linked list length field should give the total number of element descriptors on the list.

Elements of memory descriptors comprise the linked list. Each element describes the starting address and the length in bytes of the memory block. The sum of the byte count of all the elements in the linked list must equal the sector count times the sector size in bytes.

8.8.2 Setting Up a Scatter/Gather Transfer (continued)

The IOPB and Linked List in Figure 8-7 illustrate a Read Transfer to 6 blocks of memory. The sector size in this cases 528-bytes per sector; we are transferring 3 sectors of information. The 732 transfers the first 16 bytes of data from each sector to a separate data buffer. It scatters the bulk of the data, 512-bytes per sector, into memory as 3 blocks having 512 bytes each.

Set SGM and execute this IOPB.

FIGURE 8-7. SCATTER/GATHER TRANSFERS

8.8.3 732 Operation

The 732 proceeds as if doing a normal read until it starts the data transfer into memory. The contents of the linked list now controls the DMA processor; it gives the processor the byte count and address for each element on the list. The processor takes the data out of the FIFO and transfers it to memory as decribed in each element on the list.

8.9 DMA THROTTLE / THROTTLE DEAD TIME

The 732 always transfers IOPBs in Word mode; it uses the last specified values for the throttle and throttle dead time.

Host software can set the Throttle Dead Time (TDT) field in the Controller Parameters IOPB. This value defines the time that the 732 waits before attempting to regain control of the bus between throttle bursts. There are four valid TDT values.

TDT VALUE	TIME
0	0 microseconds
1	3.2 "
2	6.4 "
3	12.8 "

TABLE 8-4. THROTTLE DEAD TIME VALUES

8.10 BLACK HOLE TRANSFERS

Sometimes the data to be transferred has to go to a single memory location. This single location is usually a graphics controller with a single port on the bus. The normal DMA mode increments the bus address on each transfer so the data is put into contiguous memory space. When Black Hole Transfers are implemented, the 732 does not increment the bus address between each data transfer.

Any transfer that includes a DMA to a single location should have BHT set in Byte 5 of the IOPB. This causes only the data transfer portion of the command to not have its bus address incremented. The IOPB DMA still occurs in Normal mode (i.e., the 732 increments the address).

The data address must be properly aligned: word aligned for word transfers, and longword aligned for longword transfers. The 732 cannot do dynamic mode switching with this option.

8.11 IOPB CHECKSUM

While debugging the driver, you may choose to append the checksum to the IOPB. The checksum is the sum of Bytes 0 through 17 in the IOPB, and is expressed as a 16-bit quantity. The 732 generates a checksum with the data from the IOPB and compares it to the appended checksum; a miscompare causes a fatal error. If AUD and ICS are set, the 732 appends a new checksum as it updates the IOPB. If you want to disable the checksum, the Write Controller Parameters IOPB must have a valid checksum.

8.12 INTERRUPT AT END OF CHAIN

IEC prevents the 732 from interrupting after completing each IOPB in a chain. The 732 executes the entire chain and then interrupts (using the interrupt level and vector from the first IOPB in the chain). When IEC is clear, the 732 interrupts after completing each IOPB (providing the interrupt level is not zero).

8.13 RELEASE ON REQUEST

When ROR is enabled, the 732 tests the VMEbus between each throttle for other pending bus requests. If another request is pending, the 732 releases the bus. If there are no bus requests, the 732 remains bus master. The throttle value determines how often the 732 tests the bus. Using lower throttle values causes the DMA to slow down; using higher throttle values causes the 732 to test the bus less frequently.