FDC-2

A Double Density Floppy Disk Controller for the S-100 Bus

Edition 2

ITHACA INTERSYSTEMS LIMITED WARRANTY

All equipment manufactured by ITHACA INTERSYSTEMS shall be guaranteed against defects in materials and workmanship for a period of ninety (90) days from date of delivery to the Buyer by the Seller, and the Seller agrees to repair or replace, at its sole option, any part which proves to be defective and attributable to any defect in materials or workmanship.

> EXCEPT FOR THE WARRANTIES THAT THE GOODS ARE MADE IN A WORKMANLIKE MANNER AND IN ACCORDANCE WITH THE SPECIFICATIONS SUPPLIED, SELLER MAKES NO WARRANTY EXPRESS OR IMPLIED, AND ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE WHICH EXCEEDS THE FOREGOING WARRANTY IS HEREBY DISCLAIMED BY SELLER AND EXCLUDED FROM ANY AGREEMENT.

Buyer expressly waives its rights to any consequential damages, loss or expense arising in connection with the use of or the inability to use its goods for any purpose whatsoever.

No warranty shall be applicable to any damages arising out of any act of the Buyer, his employees, agents, patrons or other persons.

In the event that a unit proves to be defective, and after authorization by Seller, the defective part and/or unit, as authorized, must be securely packaged and returned Freight Prepaid by the Buyer to ITHACA INTERSYSTEMS for repair. Upon receipt of the unit, ITHACA INTERSYSTEMS will repair or replace, at its sole option, the defective part or product and return such part/product Freight Prepaid to the Buyer.

The remedies set forth herein are exclusive and the liability of Seller to any contract or sale or anything done in connection therewith, whether in contract, in tort, under any warranty, or otherwise, shall not, except as expressly provided herein, exceed the price of the equipment or part on which said liability is based.

This warranty is given solely to the original Buyer. No employee or representative of Seller is authorized to change this warranty in any way or grant any other guaranty or warranty.

FDC-2

A Double Density Floppy Disk Controller
for the IEEE 696.2 S-100 Bus

Economical, flexible, efficient mass storage has become a basic requirement for all but the most primitive microcomputer systems. Floppy disks are a recognized standard for this purpose, providing users with not only a means for storage, but for communication as well. The FDC-2 floppy disk controller implements this fundamental mass storage function reliably and elegantly.

Double density, dual head capacity brings large data base processing into the reach of the small system, while direct memory access means that the controller is entirely responsible for data transfer, eliminating the need for wait states and other processing required in less sophisticated floppy disk systems. Conforming to the new IEEE 696.2 S-100 standard, including 24-bit extended addressing DMA, the FDC-2 is entirely upward compatible with the new 16-bit processors, such as Ithaca Intersystems' Series II MPU-8000 Z-8000 CPU card.

The FDC-2 is available with the CP/M™ operating system, for use in standard 8-bit microcomputers. With an Ithaca Intersystems MPU-80 Z-80 card and IEEE S-100 extended addressing memory, 8080/Z80 application programs can use a megabyte of address space, all of which can be directly loaded by the FDC-2 floppy disk controller (under application program control).

Adaptable to 5-1/4" and 8" drives, compatible with old and new S-100 systems, the FDC-2 provides the microcomputer designer and user with a mass storage interface for most current and anticipated floppy disk applications.

TABLE OF CONTENTS

Section 1

Introduction and General Information

1.1 Service Information

Receiving Inspection
Factory Service
Contacting Intersystems

1.2 Overview

Disk Controller
DMA Controller
Addressing
Onboard EPROM

1.3 FDC-2 Address Map

1.4 Floppy Disks and Floppy Disk Operating Systems

Operating Systems

1.0 Introduction and General Information

The Intersystems Floppy Disk Controller (FDC-2) is a powerful and versatile mass storage system for the S-100 computer. The FDC-2 offers the following features:

* Up to 4 Mbyte direct access mass storage
* Single or Double Density
* Soft Sector IBM compatible
* 8" or 5.25" disks, single or double sided
* 1 thru 4 drives controlled by one board
* LSI controller with extensive instruction set including disk to memory compare instructions
* DMA data transfers between disk and system memory relieve CPU of cumbersome transfer routines
* Efficient operation allows entire track contents transferred in one disk revolution
* On board EPROM
* CPM compatible
* IEEE 696.2 S-100 compatible including:
  2 or 4 MHz operation;
  8 or 16 bit I/O mapping;
  16 or 24 bit memory mapping for EPROM;
  16 or 24 bit DMA.

1.1 Service Information

Receiving Inspection

When your FDC-2 arrives, inspect both the equipment and the shipping carton immediately for evidence of damage during transit. If the shipping carton is damaged or water-stained, request the carrier's agent to be present when the carton is opened. If the carrier's agent is not present when the carton is opened, and the contents of the carton are damaged, save the carton and packing material for the agent's inspection. Shipping damages should be immediately reported to the carrier. Do not attempt to service the board yourself as this will void the warranty.

We advise that in any case you should save the shipping container for use in returning the module to Intersystems, should it become necessary to do so.

Factory Service

Intersystems provides a factory repair service for all of its products. Before returning the module to Intersystems, first obtain a Return Authorization Number from our sales department. This may be done by calling us, sending us a TWX, or by writing to us. After the return has been authorized, proceed as follows:

1) Write a letter describing the problem as best you can.

2) Describe your system to us, list boards by manufacturer and name.

3) Include Xerox copies of the schematics of boards by manufacturers other than Intersystems.

4) Include the Return Authorization Number.

5) Pack the above information in a container suitable to the method of shipment.

6) Ship prepaid to Intersystems.

Your module will be repaired as soon as possible after receipt and return shipped to you prepaid.

Contacting Intersystems:

The following apply both for correspondence and service.

Ithaca Intersystems Inc.
1650 Hanshaw Rd.
P.O. Box 91
Ithaca N.Y. U.S.A.
14850

Telephone (607) 257-0190
TWX 510 255-4346

In Europe:

Ithaca Intersystems (U.K.) Ltd.
58 Crouch Hall Rd.
London N8 8HG. U.K.

Telephone 01-341-2447
Telex 299568

1.2 Overview

The FDC-2 may be instructed, through a series of output operations, to perform two basic functions: read data from anywhere in memory and write it at a desired drive, side, sector, and track; and the inverse function of reading specified disk data and writing it at any desired location in system memory.

The FDC-2 provides a modern disk interface for the S-100 bus. A single 8", double density, single sided disk provides the user with approximately 500 kbytes of mass storage. Since the controller board is capable of handling 4 double sided disks, 4 Mbytes of mass storage is available.

The FDC-2 may be parsed out into separate functional areas for easier understanding. These are:

Disk Controller

This area of the FDC-2 interfaces directly with the disk drives. It can be instructed by the CPU to select one of four drives, move the recording head in a specified drive to a specified track position, read a specified quantity of data from a drive, write a quantity of data to a drive, format a disk, and even compare data on the disk to system data. The controller also performs CRC checks on all disk data to establish that data's validity.

DMA Controller

This functional area of the FDC-2 is dedicated to the direct transfer of data between the Disk Controller and the user's system memory. The DMA controller has the ability to disable the system CPU asynchronously to program execution, whenever the Disk Controller indicates readiness for data transfer. Once the CPU is disabled the DMA Controller will either read a byte from system memory and send it to the Disk controller, or read a byte from the Disk Controller and write it into system memory. After the DMA transfer, the CPU is re-enabled and will continue program execution. The user may program the DMA Controller to access any area of an IEEE 696.2 S-100 16 Mbyte address space.

Addressing

The FDC-2 occupies 16 consecutive locations of system I/O space. The function assignment of each of these locations is shown in the FDC-2 Address Map. The user may jumper select the board location at any 16-location boundary in the standard IEEE S-100 64k I/O space. Optionally, of course, the 8-bit I/O address space may be referenced instead.

Onboard EPROM

Provision is made for a 2708 EPROM on the FDC-2 so that bootstrap firmware for the user's operating system may be located on the board. The EPROM circuitry has its own independent address. The user may jumper-select any 1 kbyte location in the extended 16 Mbyte system memory to locate the EPROM. The board may be configured so that PHANTOM is driven when the EPROM is accessed, thus allowing the FDC-2 EPROM to overlay RAM memory that responds to PHANTOM; the EPROM -- and the PHANTOM overlaying signal -- may be disabled with software. (The EPROM is enabled automatically when power is applied and RESET* goes active.)

1.3 FDC-2 Address Map

Base address is set to 00H.

Address (hex)		Function	
Base:	00	Disk Controller Status/Command	(R/W)
	01	Disk Controller Data	(R/W)
	02, 03	set DMA write to system memory*	(W)
	04, 05	set DMA read from system memory*	(W)
	06, 07	enable EPROM*	(W)
	08, 09	DMA upper address byte*	(W)
	0A, 0B	DMA middle address byte*	(W)
	0C, 0D	DMA lower address byte*	(W)
	0E, 0F	disable EPROM*	(W)

* In each of these cases, either port listed will perform the specified function and may be used interchangeably; that is, the FDC-2 disregards A0.

1.4 Floppy Disks and Floppy Disk Operating Systems

A floppy disk is a round piece of plastic mylar coated with a magnetic recording film -- much like the material magnetic recording tape is made out of, only stiffer and shaped like a phonograph record. A floppy disk, in fact, has some of the advantages of a phonograph record: access to any part of the disk is relatively quick, much like picking up the needle of a phonograph record and moving it to another band. This is inherently faster than reel-to-reel tape or cassettes, where it is necessary to run through the reel to get to a particular piece of data.

The plastic disk is permanently encased in a square paper or cardboard jacket. It is never removed from this jacket. The disk may be inserted into a disk drive which rotates the disk inside the jacket. The drive also contains a high quality playback/recording head which can be loaded into contact with the disk. The head is affixed to some sort of mechanism that allows it to move along the radius of the disk. The force for this movement is usually provided by a stepper motor, so that the head is moved incrementally from one "track" to another. On an 8" disk, there are 76 of these tracks available for data storage. On such a disk, roughly 6.5 kbytes can be stored serially on a single "track" when recorded in double density. Usually, this data is organized in smaller "sectors": angular portions of the whole track. Various systems use various numbers of sectors; a standard CP/M™ single density disk has 26 sectors.

Operating Systems

The term "operating system" (OS) refers, broadly, to any software that allows a human being to interface with a computer system. A floppy disk operating system obviously implies interface to a disk device as well as the computer.

The operating system is the thing that prints the asterisk, period, or other prompt character on the terminal when the user first turns the computer on. This is one of the customary tasks of an OS: it is the first major program to get control. (Most disk operating systems actually have one or more "bootstrap" programs that initially load the operating system into memory before passing control to it.)

The EPROM monitors that some computer companies market are, in fact, simple operating systems. A simple program of this kind allows the user to communicate with the computer at a relatively primitive level, usually providing load and dump memory instructions so that programs may be loaded directly in machine language. The monitor provides a basic communication facility by containing software that handles a terminal -- where the user types his instructions to the monitor and reads the data the monitor gets.

A disk operating system is an elaborate monitor. It also provides the basic function of terminal communication with the user. In addition, of course, it provides facilities for getting and loading data to and from disks. Other features are provided by various operating systems, from assembler software (a program that will translate a file filled with symbolic assembly code written by the user into machine code suitable for execution) to facilities for multi-tasking, various high-level languages, and so forth.

Operating systems are commonly designed to be modular: that is, programs can be added later on, either by the user or by the manufacturer. The operating system will often be provided with a few modular parts already included, usually the most-needed basic programs, and these programs are referred to as "utility programs" or just "utilities". A program that prints a specified disk file on the system line printer, for instance, would be a utility. Various operating systems assign different functions to the OS directly, leaving other functions to the utilities. One OS, for instance, might provide the listing function as an OS instruction; another would require that a utility program be loaded first to execute the same kind of function.

One extremely popular operating system is the CP/M™ system. The advantages of using CP/M™ can really be summed up in one word: compatibility. Programs that operate in a CP/M™ environment are available from many different sources, and users of CP/M™ can exchange data and user-written programs with ease, merely by trading disks.

Although all standard CP/M™ disk files are compatible, all CP/M operating systems are by no means identical. Each CP/M™ system has certain individual characteristics. One of these is memory size: a program designed to run on a large CP/M™ system will not execute successfully on a small one, because there is simply not enough memory. The other significant variable is the input/output (I/O). The CP/M™ manual gives more details on this topic, but suffice it to say that every different piece of I/O hardware requires a corresponding adjustment in the operating system; one manufacturer's disk controller will require different software than another's, and the same is true of different terminal interfaces (although the modification of the terminal handling part of the code is usually much easier than changing the disk handling portion). Ithaca Intersystems, of course, offers a version of CP/M™ that runs with the FDC-2 Disk Controller Board.

Section 2

Board Setup

2.1 I/O Addressing

2.2 EPROM Addressing

2.3 Wait States

2.4 Interrupts

2.5 DMA

2.6 Standard or Mini Disk Drive

2.7 Precompensation

2.8 Notes on Drive Configuration
- Stepper Motor Enable
- Head Load
- Multiple Drives

2.9 Standard Ithaca Intersystems Board Setup

2.0 Board Setup

The user should select and check the various jumper selectable options on the FDC-2 board before inserting the board into the user's S-100 system. The jumper selectable options include:

* I/O Addressing
* EPROM Addressing
* Wait State Selection
* System Interrupt Line Selection
* DMA Configuration
* 8" or 5.25" Disk Drives
* Precompensation Values

2.1 I/O Addressing

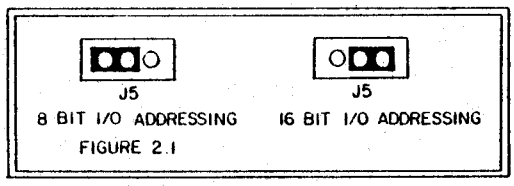


FIGURE 2.1

The user should select the board I/O address to correspond with the system hardware and software. First, the user should set the board for either 8 bit or 16 bit I/O addressing. This decision should be based upon what I/O addressing the system CPU uses. Shunt jumper J5 selects between 8 or 16 bit I/O addressing. See Figure 2.1.

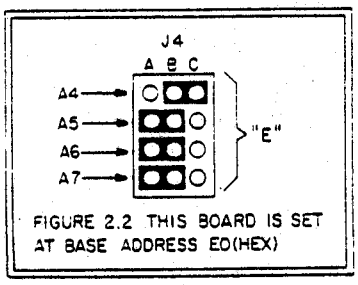


FIGURE 2.2 THIS BOARD IS SET AT BASE ADDRESS E0(HEX)

Secondly, the user should select the board I/O address. If the board is 8 bit addressed, then only jumper area J4 affects board address. The user may select address bits 4 through 7. See Figure 2.2.

Finally, if the board is 16 bit I/O addressed, then jumper area J19 affects the upper 8 bits of board address, A8 through A15. See Figure 2.3.

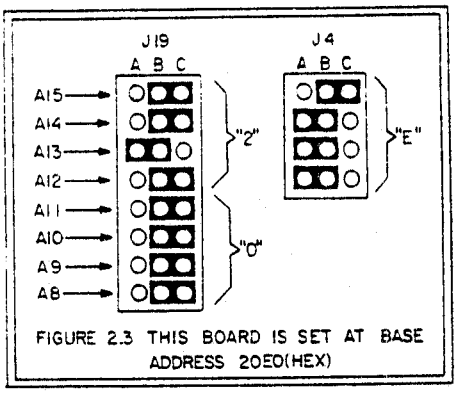


FIGURE 2.3 THIS BOARD IS SET AT BASE ADDRESS 20E0(HEX)

2.2 EPROM Addressing

The user must set the EPROM memory address to correspond to system hardware and software. Jumper area J20 determines the enabling and the base address. The options are:

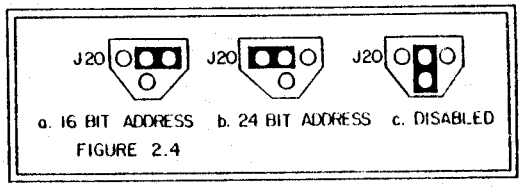


FIGURE 2.4

* The EPROM is enabled and has a 16 bit base address.
* The EPROM is enabled and has a 24 bit base address.
* The EPROM is disabled.

See Figure 2.4.

The EPROM address itself is selected at jumper area J21. For 16 bit addressing, J21 is used to set address bits 10 to 15. See Figure 2.5.

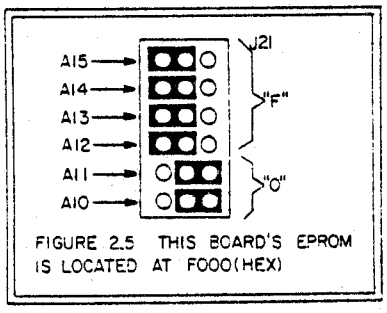


FIGURE 2.5 THIS BOARD'S EPROM IS LOCATED AT F000(HEX)

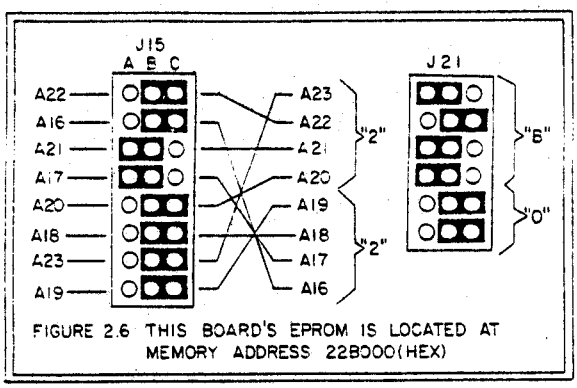


FIGURE 2.6 THIS BOARD'S EPROM IS LOCATED AT MEMORY ADDRESS 228000(HEX)

If 24-bit addressing is being used in the system, then J15 must also be set to correspond to address bits A16 through A23. See Figure 2.6.

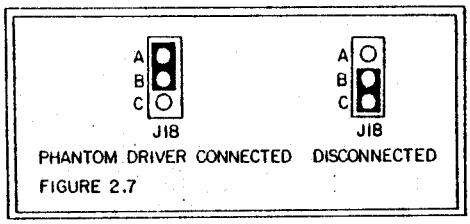


FIGURE 2.7

Finally, the user may decide if the EPROM will be phantom memory or not. If the EPROM is phantom, it will drive the S-100 phantom line when it is selected, disabling any other memory that may happen to be located at the same address, avoiding conflicts on the bus that might occur between the EPROM and read/write memory. Note that the FDC-2 EPROM may be disabled by software after power on (the reset which occurs at power on automatically enables the EPROM): writing to a port on the board provided for this purpose disables the EPROM, while writing to an additional port provided will subsequently enable it, if desired (see the FDC-2 address map in section 1). This allows the EPROM to be used for a bootstrap load of the operating system after which it may be turned off, freeing that space for RAM. To enable the phantom driver, use J18 as shown in Figure 2.7.

2.3 Wait States

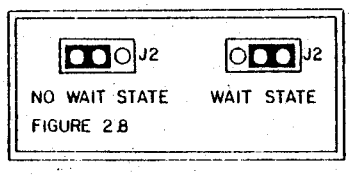


FIGURE 2.8

When running in a 4 MHz system the EPROM and the LSI disk controller may not have sufficient access time for valid data transfers with the CPU. If this is the case the user should set jumper area J2 so that the FDC-2 generates one wait state at each access. See Figure 2.8.

If the board is set to generate a wait state at access, then the polarity of the system clock Phi becomes significant. Jumper area J14 provides for either the IEEE 696.2 S-100 standard clock, or a non-standard inverted clock. See Figure 2.9.

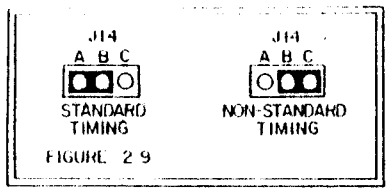


FIGURE 2.9

2.4 Interrupts

The FDC-2 generates an interrupt signal at the end of various procedures so that the system CPU can respond to the event with appropriate action. The S-100 bus provides 10 interrupt lines: NMI, pINT, and the vectored interrupt lines VI0 through VI7. The FDC-2 interrupt signal may be jumpered to any one of these, with the exception of NMI (which is customarily reserved for emergency system functions such as power failure processing).

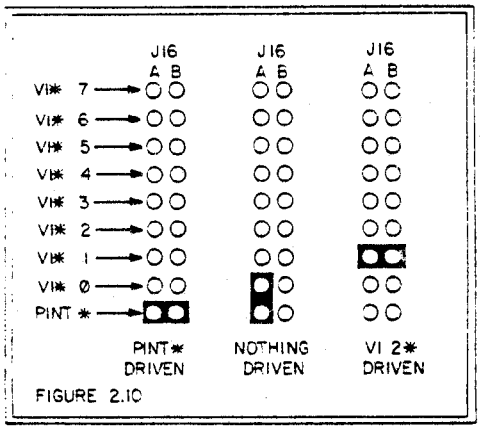


FIGURE 2.10

If there is a system interrupt controller (on the CPU card, perhaps) the FDC-2 and other system peripherals might drive one or another of the vectored interrupt lines. Alternately, in a system without an interrupt controller the FDC-2 can be jumpered to drive pINT. In either case, of course, software must be available to deal with interrupts. In the case of vectored interrupts, generally the interrupt controller device is initialized, and interrupt service routine address vectors stored there. In the case of a pINT system, care must be taken that no other device in the system drives the pINT line, and that no device in the system responds to the interrupt acknolwedge signal, sINTA. If these conditions are met, then the FDC-2's interrupt will cause the CPU to read and execute a byte of FFH, which, in 8080 code, is a RST 38H. Alternately, a Z80 processor may be set to Interrupt Mode 1, in which case interrupts automatically generate a RST 38H, regardless of the response byte. Location 38H would then contain the interrupt service routine or a jump to one. In a minimal system, the CPU could enter a halt state after it initiates a disk operation; in this case, the interrupt routine at 38H could simply be a RET instruction.

An entirely different -- and in many cases, simpler -- approach to software response to disk board activities is the use of the Polled Mode. In this case, the FDC-2 drives no interrupt line at all, and the CPU executes a short polling routine while floppy disk operations are carried on.

Jumper area J16 is provided to select which system interrupt line -- if any -- will be driven by the FDC-2. See Figure 2.10. In the event that no interrupt line is to be driven, the shorting strip may be stored on any two pins in column A of J16.

2.5 DMA

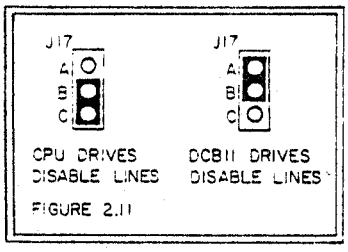


FIGURE 2.11

In some systems, the DMA disable lines -- ADSB, DODSB, SDSB, and CDSB -- will be driven by the CPU card. Other systems expect the DMA device to drive these lines. The FDC-2 makes provision for both possibilities at jumper area J17. See Figure 2.11.

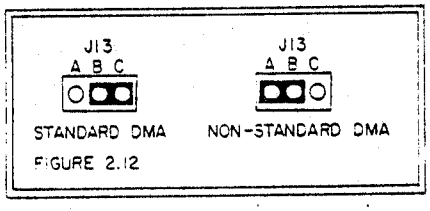


FIGURE 2.12

The IEEE 696.2 S-100 standard provides for overlapped transfer at either end of a DMA operation. Pre-standard CPUs may instead conduct an immediate transfer. To adapt to these differences the FDC-2 may be jumpered at J13 to either turn off the hold signal at the standard point of the DMA cycle, or to extend the hold signal one half clock cycle, to conform to non-standard CPUs. See Figure 2.12.

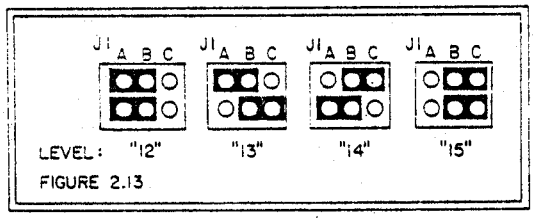


FIGURE 2.13

The standard also provides for 16-level arbitration among DMA devices. The FDC-2 provides the user with the option to occupy any one of the top four DMA priority levels -- levels 12 through 15. The priority level may be selected at jumper area J1, as depicted in Figure 2.13.

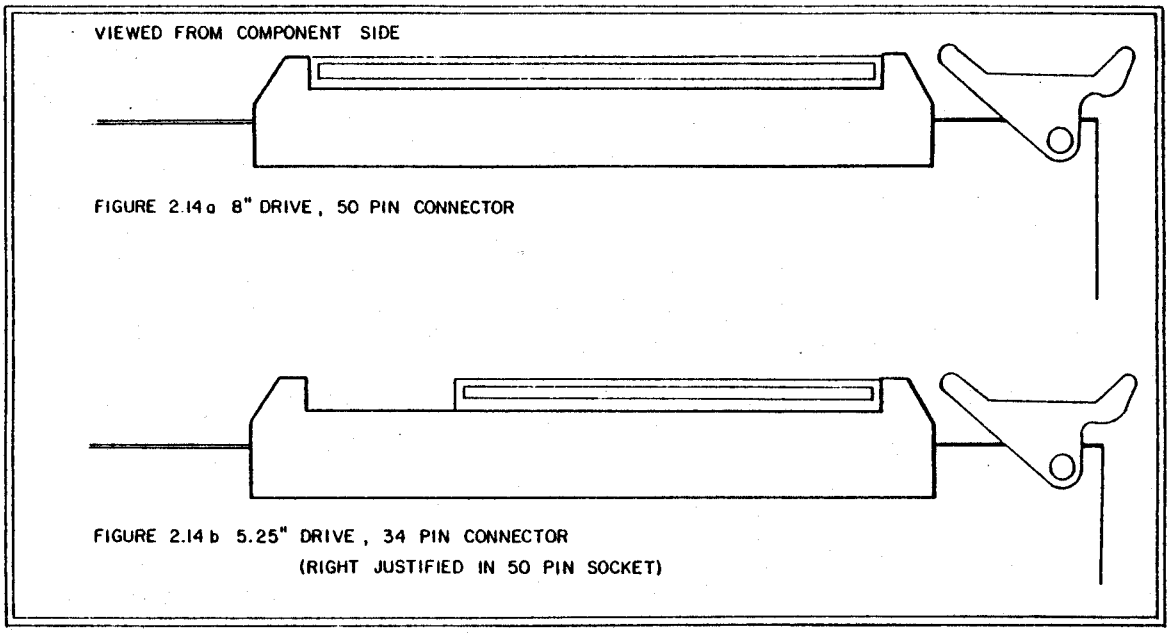


FIGURE 2.14a 8" DRIVE, 50 PIN CONNECTOR

FIGURE 2.14b 5.25" DRIVE, 34 PIN CONNECTOR
(RIGHT JUSTIFIED IN 50 PIN SOCKET)

2.6 Standard or Mini Disk Drive

The FDC-2 can control 8" disk drives or 5.25" disk drives. The 50-pin header on the top right of the board will accept the standard 50-pin connector for the 8" drive or the standard 34-pin connector right justified in the header for the 5-1/4" drive. See Figure 2.14

Various signals must be adjusted to correspond to 8" or 5-1/4" drives. These should be set by the user at jumper areas J6 through J12 as shown in Figure 2.15.

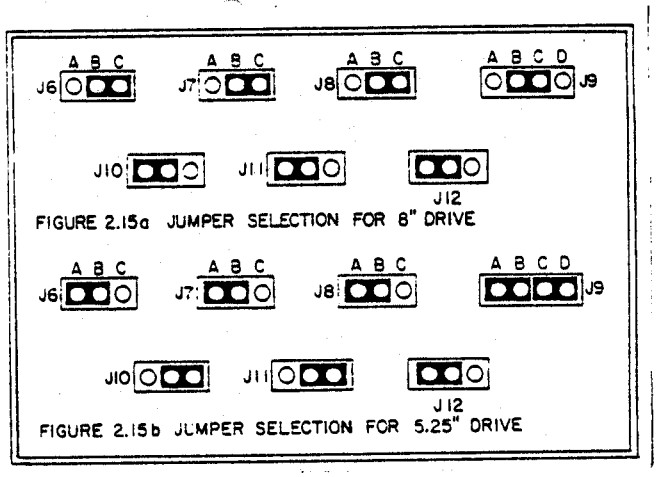


FIGURE 2.15a JUMPER SELECTION FOR 8" DRIVE

FIGURE 2.15b JUMPER SELECTION FOR 5.25" DRIVE

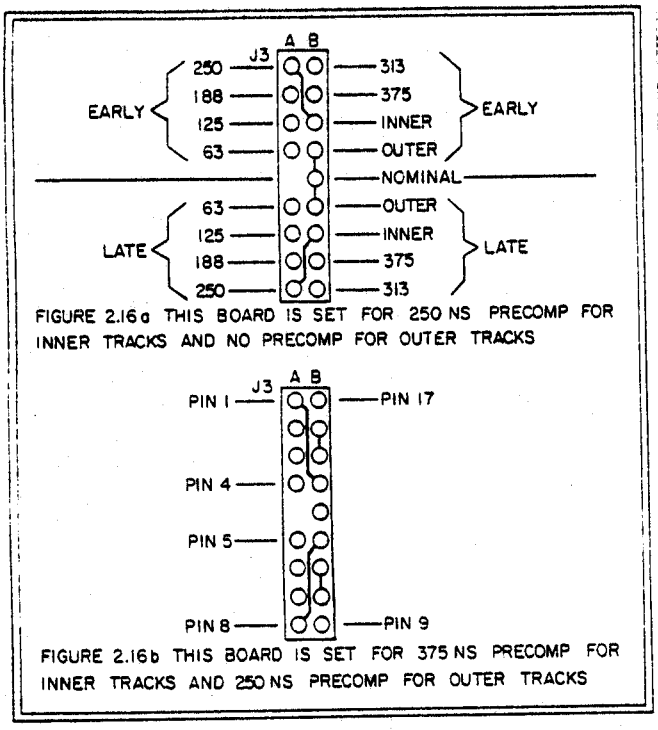


FIGURE 2.16a THIS BOARD IS SET FOR 250 NS PRECOMP FOR INNER TRACKS AND NO PRECOMP FOR OUTER TRACKS

FIGURE 2.16b THIS BOARD IS SET FOR 375 NS PRECOMP FOR INNER TRACKS AND 250 NS PRECOMP FOR OUTER TRACKS

2.7 Precompensation

Precompensation is a slight shift that is imposed on the position of data pulses as they are written on the disk. This is done to compensate for expected shift when data is read back, due to the resolution of the drive head.

Drive manufacturers provide precompensation values to be used with their drives for double density data. The FDC-2 may be jumpered for six different values from 62 to 375 ns. The user may also set one value for inside tracks, and a different value for outside tracks. This can provide additional reliability with many drives. Figure 2.16, a and b, shows various precompensation options at jumper area J3.

2.8 Notes on Drive Configuration

User-selectable options of various manufacturers' floppy disk drives must be set correctly to insure dependable operation with the FDC-2 controller.

Stepper Motor Enable

For correct operation with the FDC-2, the floppy disk drive should be jumpered to have a continuously enabled stepper motor. The stepper motor should NOT require active drive select, and/or head load, in order to be enabled.

Stepper Motor Enable	
Shugart 800/850, Remex 4000, Qume	Jumper HL: Open DS: Open

Head Load

The floppy disk drive should be jumpered to load the heads on active head load alone. The drive should NOT require active drive select to load the heads.

Head Load	
Shugart 800/850, Remex 4000, Qume	Jumper C: Closed X: Closed A: Closed B: Open

Multiple Drives

The FDC-2 (and most other disk controllers) require that the floppy disk drive interface signal lines should only have one pullup resistor per line. Usually this involves removing the pullup resistor pack from all but one drive in a system. This is not always true, however, and problems will sometimes arise, especially when mixing drives of different manufacturers in one system. Often shunt jumpers are provided in the drive to disconnect individual pullup resistors from control lines. Check the manufacturer's documentation carefully, particularly when mixing different manufacturers' drives. Improper termination -- when more than one resistor is pulling up a line, or if no resistor is pulling up a line -- will sometimes result in intermittent operation of the system, ranging from very occasional errors to continuously "flaky" response.

When there is a choice, it is usually good practice to terminate the drive in a multiple-drive system that is furthest -- physically, along the common connector cable -- from the FDC-2; that is, the last drive on the cable, whether it is drive A or not. This last drive, then, would be the one in which an optional resistor pack should be left installed, or the jumpers set so as to enable the pullup resistors.

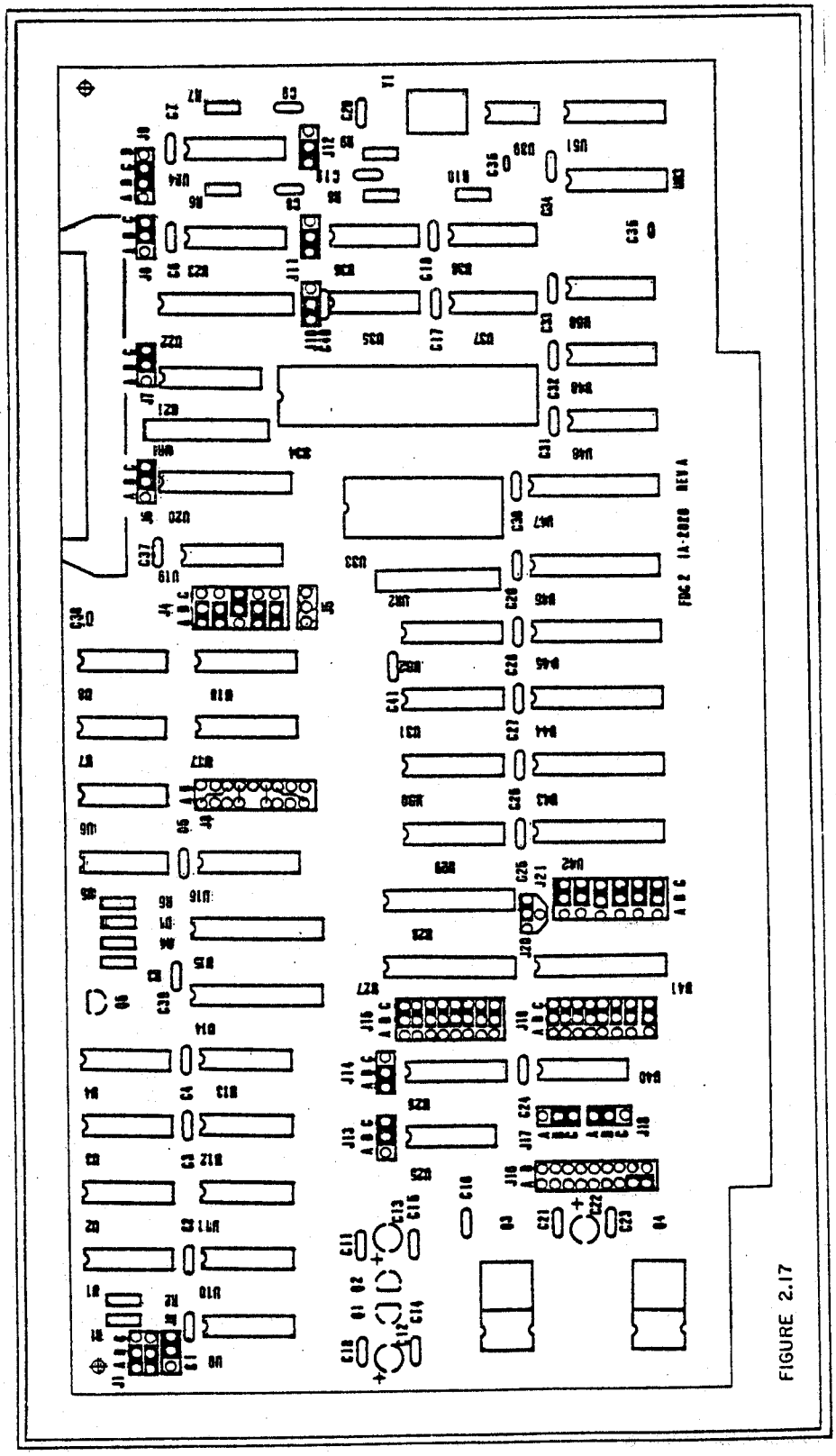


FIGURE 2.17

2.8 Standard Ithaca Intersystems Board Setup.

Figure 2.17 shows the arrangement of jumpers for the standard Ithaca Intersystems setup, as listed below.

Implemented Option	Jumper
DMA Priority 12	J1, 1st row: AB 2nd row: AB
One wait state	J2: BC
MFM Precompensation -- inner tracks: 256 ns outer tracks: 62.5 ns	J3: pins 1 to 15, pins 4 to 14, pins 5 to 12, pins 8 to 11
Board address = B0H	J4, 1st row: AB 2nd row: AB 3rd row: BC 4th row: AB
8 bit I/O Address (no extended address)	J5: AB
Standard 8" drive	J6: BC
	J7: BC
	J8: BC
	J9: BC
	J10: AB
	J11: AB
	J12: AB
Standard DMA timing	J13: BC
Wait states clocked on inverted Phi	J14: AB
EPROM extended address = 00H	J15, all rows: BC
No interrupt lines driven	J16, jumper stored on any two pins of row A
No bus disable lines driven	J17: row 2-row 3
EPROM select drives PHANTOM	J18: row 1-row 2
I/O extended address = 00H	J19, all rows: BC
EPROM enabled, 16 bit address (no extended address)	J20, 1st row: BC
EPROM address = 0000H	J21, all rows: BC

Section 3

FDC-2 Programming Guidelines

3.1 Programming the NEC uPD765

3.2 Polled Operation

3.3 Programming DMA Transfers

3.4 EPROM Programming

3.0 FDC-2 programming guidelines

In most cases the user will obtain a FDC-2 board with the operating system software at the same time, and will not be concerned with the details of the implementation. This section is for those who wish to write their own software to drive the FDC-2.

Three areas of the FDC-2 are affected by programming. These are the LSI controller chip itself, the DMA transfer circuitry, and the EPROM. Refer to the FDC-2 I/O map for address location of onboard registers.

3.1 Programming the NEC uPD765

The major part of the FDC-2 operation is conducted by the NEC uPD765, an LSI double density disk controller IC. The uPD765 has a repertoire of 15 different operations which the CPU may instruct it to perform. These are:

Read Data	Scan High or Equal	Write Deleted Data
Read ID	Scan Low or Equal	Seek
Read Deleted Data	Specify	Recalibrate
Read a Track	Write Data	Sense Interrupt Status
Scan Equal	Format Track	Sense Drive Status

The details of this intruction set are explained in the NEC uPD765 manual, to which the interested reader is referred. A quick summary is presented here to relate the uPD765 operation to the rest of the FDC-2.

As the uPD765 performs an operation, there are three phases which must be considered by the programmer. These are:

Command phase: This phase exists for all operations. In this phase the CPU sends command bytes to the uPD765 to indicate which operation is desired and the arguments of the operation (drive, track, sector, etc.). The uPD765 requires that the CPU check the main status register for an active request for master (bit 7 high) and a data input condition (bit 6 low) before each command byte is sent. When the uPD765 recognizes a complete command byte string it will automatically enter the execution phase.

Execution phase: After the uPD765 accepts the command string it will begin to perform the operation. No software interaction is necessary during this phase. If the operation encompasses data transfer between the system and the disk (during read, write, scan, or format), the DMA hardware on the FDC-2 board

conducts the transfer with system memory, and should be programmed prior to the execution phase (programming DMA is covered below). The uPD765 will generate an interrupt signal at the end of the execution phase on operations where data transfers take place as well as seek and recalibrate operations. This interrupt may be serviced by the system in any fashion suitable to the user's requirments.

Result phase: After the execution phase, most operations of the uPD765 enter the result phase. At this point, the uPD765 provides the user with status concerning the success of the operation just attemped: a string of bytes is (and must be) transferred between the uPD765 and the system CPU. These bytes are the status bytes which are described in the uPD765 manual. The CPU must check the main status register for an active master request (bit 7 high) and data in to the CPU (bit 6 high) before reading each status byte in the result phase. The uPD765 will not accept new commands until the correct number of status bytes are read to complete the result phase of the last operation.

3.2 Polled Operation

While the uPD765 is frequently used in the interrupt mode (the chip produces an interrupt to signal the completion of an activity), polled operation is also possible, and in some configurations is desirable. The uPD765 can be polled by checking various chip registers repeatedly. The following code is typical.

```
1000*1000                ORG     1000H

     00B0        DBASE   EQU     0B0H
     00B0        DSTAT   EQU     DBASE+0
     00B1        DDATA   EQU     DBASE+1
     00B2        DREAD   EQU     DBASE+2
     00B4        DWRITE  EQU     DBASE+4

                 ;THIS ROUTINE WOULD BE CALLED AFTER A DISK
                 ;READ OR WRITE HAD BEEN INITIATED.
```

```
1000 DB B0      WAITFDC IN      DSTAT   ;A GETS THE DISK STATUS.
1002 CB 67              BIT     4,A     ;TEST BIT 4 FOR ZERO.
1004 CA 100F            JZ      ERROR   ;IF BIT 4 IS NOT HIGH, THEN
                          ;THE UPD765 IS NOT EXECUTING A READ OR
                          ;WRITE AND THIS ROUTINE WAS CALLED IN ERROR.
1007 E6 C0              ANI     0C0H    ;DISCARD ALL BUT BITS 6 AND 7.
1009 FE C0              CPI     0C0H    ;TEST THEM.
100B C2 1000            JNZ     WAITFDC ;IF BOTH ARE NOT HIGH, WAIT.
                ;BIT 6 INDICATES THE DATA TRANSFER DIRECTION THE
                ;NEC 765 IS CONTEMPLATING, AND BIT 7 IS HIGH WHEN THE
                ;DISK CONTROLLER CHIP IS READY FOR DATA TRANSFER.
                ;BOTH THESE BITS WOULD BE HIGH WHEN THE NEC765 HAD
                ;COMPLETED A READ OR WRITE OPERATION AND WAS
                ;READY FOR THE RESULT PHASE.  "DATA TRANSFER"
                ;HERE REFERS TO THE PROCESS OF READING OR WRITING
                ;STATUS OR COMMANDS TO THE NEC765'S DATA
                ;REGISTER, NOT TO BE CONFUSED WITH THE ACTUAL
                ;TRANSFERS OF DATA FROM OR TO THE FLOPPY DISK,
                ;WHICH HAS PRESUMABLY BEEN ACCOMPLISHED BY DMA
                ;OPERATIONS WHILE THE WAIT LOOP HAS BEEN FUNCTIONING.
100E C9                 RET             ;OVER.
```

To poll the uPD765 while it is executing a SEEK operation, the program would continually execute a sense interrupt status command, and then check the DSTAT register for the four busy signals (by ANDing the byte with 0FH); the seek is complete when the uPD765 is no longer busy.

3.3 Programming DMA Transfers

The DMA control circuitry automatically responds to any DMA request from the uPD765 by activating the S-100 hold signal. When acknowledged by the CPU the DMA circuit will generate one S-100 memory access cycle. It is up to the user's program to establish the address to be accessed and the direction of the transfer.

There are three DMA address registers on the FDC-2, providing for 24-bit extended addresses. When written to by the user's program these registers form a DMA address pointer. The next FDC-2 DMA cycle will access this address. After a DMA cycle this pointer will be automatically incremented to point to the next memory address. The DMA circuitry will access consecutive addresses in memory until the user program again writes to the FDC-2's address registers. The FDC-2 DMA address pointer only increments the 16 least significant bits; consequently, DMA transfers are limited to 64k boundaries in system memory. The upper 8 bits of the 24 bit DMA address pointer must be incremented in software.

A DMA cycle may be one of two types -- a read or a write. In the former case, the FDC-2 reads data from system memory and sends this data to the uPD765 (usually for writing on the disk). In a DMA write, the FDC-2 gets data from the uPD765 (which is reading from the disk) and sends this data to memory. The user's program selects the type of DMA cycle. To select a read-from-memory / write-to-disk DMA cycle, the software need only execute an output instruction to the "set DMA memory read" port. To select a read-from-disk / write-to-memory cycle, the software need only execute an output instruction to the "set DMA memory write" port.

3.4 EPROM Programming

EPROM programming consists of enabling and disabling the EPROM. If the FDC-2 is shunt selected with the onboard EPROM enabled, then the EPROM will be enabled after system reset. The user's software can disable the EPROM by simply executing an output to the FDC-2 "disable EPROM" port. The software may re-enable the EPROM by simply executing an output instruction to the FDC-2 "enable EPROM" port.

Section 4

Parts List and Placement

4.0 Parts List and Placement

Figure 4.1 shows the placement of parts on the FDC-2, as specified in the following charts.

RESISTORS

Position	Value	Tolerance	Power
R1, R2	1 KOhm	10%	1/4 W
R3	2.4 KOhm	10%	1/4 W
R4	270 Ohm	10%	1/4 W
R5	470 Ohm	10%	1/4 W
R6	220 KOhm	10%	1/4 W
R7	20 KOhm	10%	1/4 W
R8, R9	330 Ohm	10%	1/4 W
R10	470 Ohm	10%	1/4 W
UR1	150 Ohm	8-pin SIP	
UR2	33 KOhm	9-pin SIP	
UR3, pins 1-16	Short (0 Ohms)		
UR3, pins 2-15	8.2 KOhm	10%	1/4 W
UR3, pins 3-14	56 Ohm	10%	1/4 W
UR3, pins 4-13	15 KOhm	10%	1/4 W

CAPACITORS

Position	Value	Type	Rating
C1 to C7, C10, C11, C14 to C18, C21 to C33, C35, C36, C37	.1 uF	Bypass	
C8	33 uF	Electrolytic	>10 V
C9	200 pF		>10 V
C12, C13	>10 uF	Tantalum	>25 V
C19	.01 uF	Bypass	
C20	10 pF	Ceramic Disk	>10 V
C22	>10 uF	Tantalum	>17 V
C34	.01 uF	DIP	
UR3, pins 7-10	120 pF		>10 V

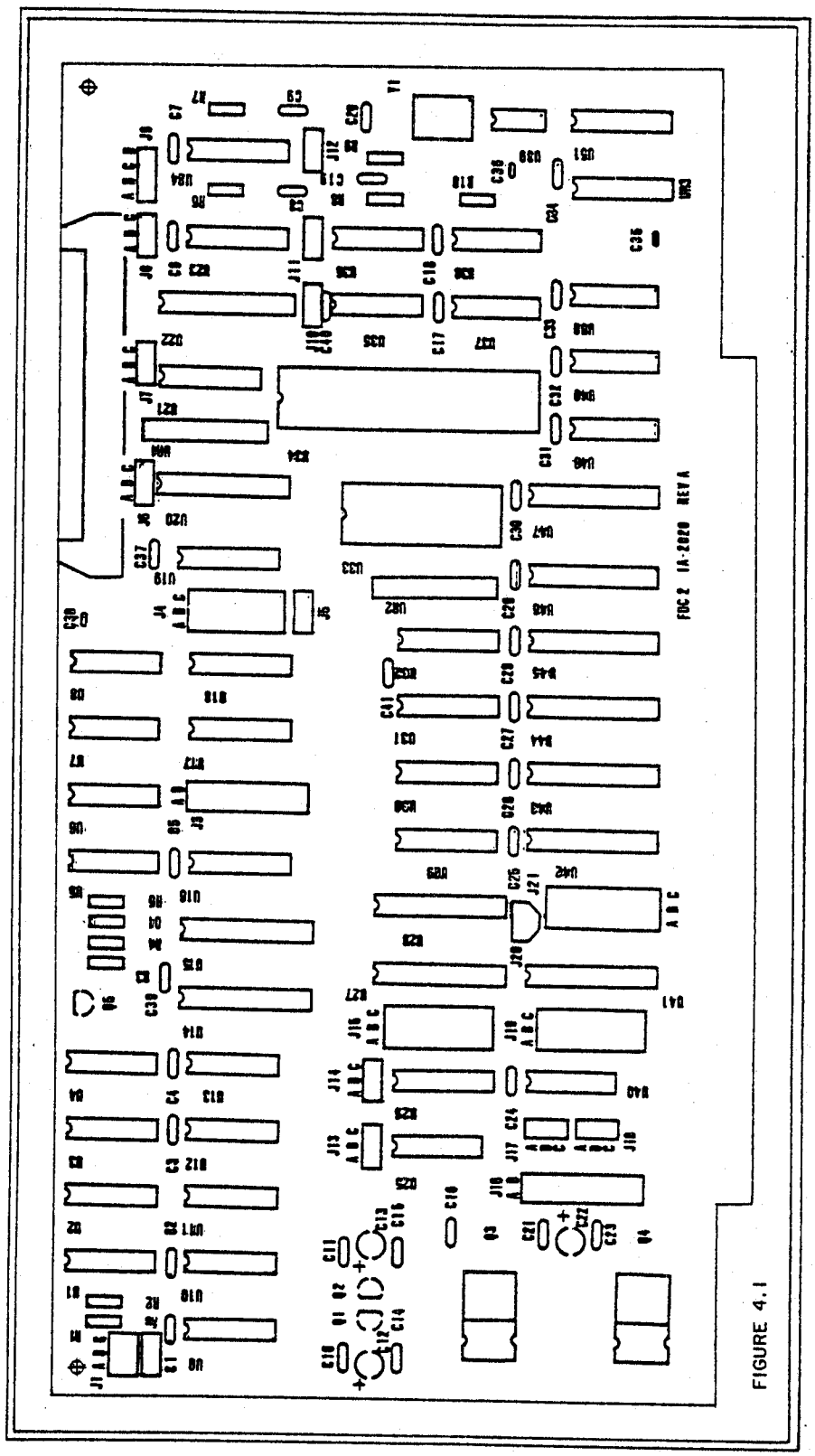


FIGURE 4.1

MISCELLANEOUS

Position	Part #	Function, Specification
Y1		16.00 MHz fundamental crystal
Q1	79L05	Low power -5 Volt regulator
Q2	78L12	Low power +12 Volt regulator
Q3, Q4	7805	+5 Volt regulator
Q5	2N2222	Transistor
D1	1N5232	5.6 Volt Zener diode

INTEGRATED CIRCUITS

Position	Part #
U1	74LS03
U2	7405
U3, U4	74LS00
U5	74LS74
U6, U7	74LS164
U8	74LS32
U9	74LS74
U10	74LS32
U11	74LS74
U12	74LS10
U13	74LS27
U14, U15	74LS240
U16	74LS157
U17	74LS151
U18	74LS155
U19	8131
U20	74S240
U21	NE590
U22	74S240
U23	74LS153
U24	96LS02
U25	74LS00
U26	74LS175
U27	25LS2521
U28	74LS373
U29, U30, U31, U32	74LS193
U33	2708
U34	uPD765
U35	74LS393
U36	74LS00
U37	74LS293
U38	74LS125
U39	CA3140
U40	7407
U41, U42	25LS2521
U43, U44	74LS244
U45, U46	8304
U47	74LS244
U48	74LS32
U49, U50	74LS74
U51	74LS124

Section 5

Revisions and Manual Applicability

5.0 Revisions and Manual Applicability

This manual references revision 0 of the FDC-2 Floppy Disk Controller Board.

Revision 0 Errors

1. DMA Error

The first ten boards of FDC-2 production contained an error in the DMA circuitry, so that the board would initiate a DMA cycle whenever the S-100 signals DMA0* and DMA1* were inactive at the same time that HOLD* and pHLDA were active, if the DMA device driving HOLD* is set to a lower DMA priority than the FDC-2. This would occur without regard to whether the FDC-2 actually needed the bus.

The error will have no effect on operation of the FDC-2 in systems where it is the only DMA device or in systems where the other DMA device or devices are set to higher priorities than the FDC-2.

The ten boards affected were all sold well before June 3, 1980.

The error is corrected by changing the board circuitry so that the board version of pHOLD* is examined rather than the S-100 bus version, by making the following cuts and jumpers:

a. On the component side of the card, cut the two traces connected to U13, pin 1. One trace goes to the left to a plated through hole; it should be cut between the plated through hole and U13, pin 1. The other trace goes to the right under the socket, and comes out between pins 13 and 14 of U13, where it may be cut.

b. JUMPER with wire wrap wire the plated through hole to the left of U13, pin 1 (mentioned above) and the plated through hole directly to the right of U14, pin 18. This reconnects the trace that was freed from U13, pin 1 in step a. (The jumper should be installed on the solder side of the card.)

c. JUMPER U13, pin 1, to U11, pin 9.

These corrections were made to all subsequent Revision 0 FDC-2 boards after the first ten, and will be incorporated in the printed circuit at Revision A.

Section 6

Ithaca Intersystems Limited Warranty

Section 7

Schematic Diagram

SINGLE/DOUBLE DENSITY FLOPPY DISK CONTROLLER

DESCRIPTION

The μPD765 is an LSI Floppy Disk Controller (FDC) Chip, which contains the circuitry and control functions for interfacing a processor to 4 Floppy Disk Drives. It is capable of supporting either IBM 3740 single density format (FM), or IBM System 34 Double Density format (MFM) including double sided recording. The μPD765 provides control signals which simplify the design of an external phase locked loop, and write precompensation circuitry. The FDC simplifies and handles most of the burdens associated with implementing a Floppy Disk Interface.

Hand-shaking signals are provided in the μPD765 which make DMA operation easy to incorporate with the aid of an external DMA Controller chip, such as the μPD8257. The FDC will operate in either DMA or Non-DMA mode. In the Non-DMA mode, the FDC generates interrupts to the processor every time a data byte is available. In the DMA mode, the processor need only load the command into the FDC and all data transfers occur under control of the μPD765 and DMA controller.

There are 15 separate commands which the μPD765 will execute. Each of these commands require multiple 8-bit bytes to fully specify the operation which the processor wishes the FDC to perform. The following commands are available:

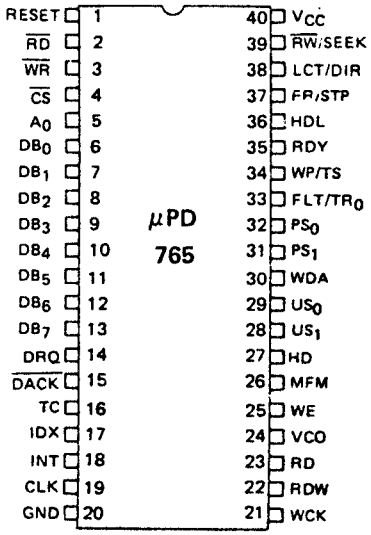
Read Data	Scan High or Equal	Write Deleted Data
Read ID	Scan Low or Equal	Seek
Read Deleted Data	Specify	Recalibrate (Restore to Track 0)
Read a Track	Write Data	Sense Interrupt Status
Scan Equal	Format a Track	Sense Drive Status

FEATURES

Address mark detection circuitry is internal to the FDC which simplifies the phase locked loop and read electronics. The track stepping rate, head load time, and head unload time may be programmed by the user. The μPD765 offers many additional features such as multiple sector transfers in both read and write with a single command, and full IBM compatibility in both single and double density modes.

- IBM Compatible in Both Single and Double Density Recording Formats
- Programmable Data Record Lengths: 128, 256, 512, or 1024 Bytes/Sector
- Multi-Sector and Multi-Track Transfer Capability
- Drive Up to 4 Floppy Disks
- Data Scan Capability – Will Scan a Single Sector or an Entire Cylinder's Worth of Data Fields, Comparing on a Byte by Byte Basis, Data in the Processor's Memory with Data Read from the Diskette
- Data Transfers in DMA or Non-DMA Mode
- Parallel Seek Operations on Up to Four Drives
- Compatible with Most Microprocessors Including 8080A, 8085A, μPD780 (Z80™)
- Single Phase 8 MHz Clock
- Single +5 Volt Power Supply
- Available in 40 Pin Plastic Dual-in-Line Package

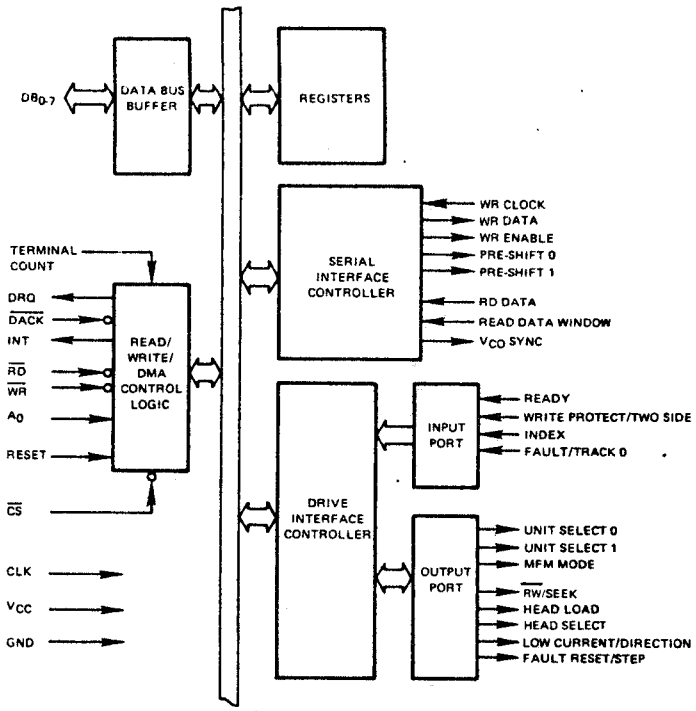
PIN CONFIGURATION



TM:Z80 is a registered trademark of Zilog, Inc.

Rev/1

BLOCK DIAGRAM



ABSOLUTE MAXIMUM RATINGS*

Operating Temperature	-10°C to +70°C
Storage Temperature	-40°C to +125°C
All Output Voltages	-0.5 to +7 Volts
All Input Voltages	-0.5 to +7 Volts
Supply Voltage V_{CC}	-0.5 to +7 Volts
Power Dissipation	1 Watt

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

*T_a = 25°C

DC CHARACTERISTICS

T_a = -10°C to +70°C; V_{CC} = +5V ± 5% unless otherwise specified.

PARAMETER	SYMBOL	LIMITS			UNIT	TEST CONDITIONS
		MIN	TYP①	MAX		
Input Low Voltage	V_{IL}	-0.5		0.8	V	
Input High Voltage	V_{IH}	2.0		V_{CC} + 0.5	V	
Output Low Voltage	V_{OL}			0.45	V	I_{OL} = 2.0 mA
Output High Voltage	V_{OH}	2.4		V_{CC}	V	I_{OH} = -200 μA
Input Low Voltage (CLK + WR Clock)	$V_{IL(\Phi)}$	-0.5		0.65	V	
Input High Voltage (CLK + WR Clock)	$V_{IH(\Phi)}$	2.4		V_{CC} + 0.5	V	
V_{CC} Supply Current	I_{CC}			150	mA	
Input Load Current (All Input Pins)	I_{LI}			10	μA	$V_{IN} = V_{CC}$
				-10	μA	V_{IN} = 0V
High Level Output Leakage Current	I_{LOH}			10	μA	$V_{OUT} = V_{CC}$
Low Level Output Leakage Current	I_{LOL}			-10	μA	V_{OUT} = +0.45V

Note: ① Typical values for T_a = 25°C and nominal supply voltage.

PIN IDENTIFICATION

PIN NO.	PIN SYMBOL	PIN NAME	INPUT/ OUTPUT	CONNECTION TO	FUNCTION
1	RST	Reset	Input	Processor	Places FDC in idle state. Resets output lines to FDD to "0" (low). Dbes not effect SRT, HUT or HLY in Specify command.
2	$\overline{RD}$	Read	Input①	Processor	Control signal for transfer of data from FDC to Data Bus, when "0" (low).
3	$\overline{WR}$	Write	Input①	Processor	Control signal for transfer of data to FDC via Data Bus, when "0" (low).
4	$\overline{CS}$	Chip Select	Input	Processor	IC selected when "0" (low), allowing $\overline{RD}$ and $\overline{WR}$ to be enabled.
5	A_0	Data/Status Reg Select	Input①	Processor	Selects Data Reg (A_0=1) or Status Reg (A_0=0) contents of the FDC to be sent to Data Bus.
6-13	DB_0-DB_7	Data Bus	Input/① Output	Processor	Bi-Directional 8-Bit Data Bus.
14	DRQ	Data DMA Request	Output	DMA	DMA Request is being made by FDC when DRQ="1".
15	DACK	DMA Acknowledge	Input	DMA	DMA cycle is active when "0" (low) and Controller is performing DMA transfer.
16	TC	Terminal Count	Input	DMA	Indicates the termination of a DMA transfer when "1" (high).
17	IDX	Index	Input	FDD	Indicates the beginning of a disk track.
18	INT	Interrupt	Output	Processor	Interrupt Request Generated by FDC.
19	CLK	Clock	Input		Single Phase 8 MHz Squarewave Clock.
20	GND	Ground			D.C. Power Return.
21	WCK	Write Clock	Input		Write data rate to FDD. FM = 500 kHz, MFM = 1 MHz, with a pulse width of 250 ns for both FM and MFM.
22	RDW	Read Data Window	Input	Phase Lock Loop	Generated by PLL, and used to sample data from FDD.
23	RDD	Read Data	Input	FDD	Read data from FDD, containing clock and data bits.
24	VCO	VCO Sync	Output	Phase Lock Loop	Inhibits VCO in PLL when "0" (low), enables VCO when "1".
25	WE	Write Enable	Output	FDD	Enables write data into FDD.
26	MFM	MFM Mode	Output	Phase Lock Loop	MFM mode when "1", FM mode when "0".
27	HD	Head Select	Output	FDD	Head 1 selected when "1" (high), Head 0 selected when "0" (low).
28,29	US_1,US_0	Unit Select	Output	FDD	FDD Unit Selected.
30	WDA	Write Data	Output	FDD	Serial clock and data bits to FDD.
31,32	PS_1,PS_0	Precompensation (pre-shift)	Output	FDD	Write precompensation status during MFM mode. Determines early, late, and normal times.
33	FLT/TR_0	Fault/Track 0	Input	FDD	Senses FDD fault condition, in Read/ Write mode; and Track 0 condition in Seek mode.
34	WP/TS	Write Protect/ Two-Side	Input	FDD	Senses Write Protect status in Read/Write mode; and Two Side Media in Seek mode.
35	RDY	Ready	Input	FDD	Indicates FDD is ready to send or receive data.
36	HDL	Head Load	Output	FDD	Command which causes read/write head in FDD to contact diskette.
37	FR/STP	Flt Reset/Step	Output	FDD	Resets fault F.F. in FDD in Read/Write mode, contains step pulses to move head to another cylinder in Seek mode.
38	LCT/DIR	Low Current/ Direction	Output	FDD	Lowers Write current on inner tracks in Read/Write mode, determines direction head will step in Seek mode. A fault reset pulse is issued at the beginning of each Read or Write command prior to the occurrence of the Head Load signal.
39	$\overline{RW}$/SEEK	Read Write/SEEK	Output	FDD	When "1" (high) Seek mode selected and when "0" (low) Read/Write mode selected.
40	V_{CC}	+5V			D.C. Power.

Note: 1 Disabled when CS = 1.

CAPACITANCE

$T_a = 25°C$; $f_c = 1$ MHz; $V_{CC} = 0V$

PARAMETER	SYMBOL	LIMITS MIN	LIMITS TYP	LIMITS MAX	UNIT	TEST CONDITIONS
Clock Input Capacitance	$C_{IN(\Phi)}$			20	pF	All Pins Except Pin Under Test Tied to AC Ground
Input Capacitance	C_{IN}			10	pF	
Output Capacitance	C_{OUT}			20	pF	

AC CHARACTERISTICS

T_a = -10°C to +70°C; V_{CC} = +5V ± 5% unless otherwise specified.

PARAMETER	SYMBOL	LIMITS MIN	LIMITS TYP①	LIMITS MAX	UNIT	TEST CONDITIONS
Clock Period	Φ_{CY}	120	125	500	ns	
Clock Active (High)	Φ_0	40			ns	
Clock Rise Time	Φ_r			20	ns	
Clock Fall Time	Φ_f			20	ns	
A_0, $\overline{CS}$, $\overline{DACK}$ Set Up Time to $\overline{RD}$ ↓	T_{AR}	0			ns	
A_0, $\overline{CS}$, $\overline{DACK}$ Hold Time from $\overline{RD}$ ↑	T_{RA}	0			ns	
$\overline{RD}$ Width	T_{RR}	250			ns	
Data Access Time from $\overline{RD}$ ↓	T_{RD}			200	ns	
DB to Float Delay Time from $\overline{RD}$ ↑	T_{DF}	20		100	ns	
A_0, $\overline{CS}$, $\overline{DACK}$ Set Up Time to $\overline{WR}$ ↓	T_{AW}	0			ns	
A_0, $\overline{CS}$, $\overline{DACK}$ Hold Time to $\overline{WR}$ ↑	T_{WA}	0			ns	
$\overline{WR}$ Width	T_{WW}	250			ns	
Data Set Up Time to $\overline{WR}$ ↑	T_{DW}	150			ns	
Data Hold Time from $\overline{WR}$ ↑	T_{WD}	5			ns	
INT Delay Time from $\overline{RD}$ ↑	T_{RI}			500	ns	
INT Delay Time from $\overline{WR}$ ↑	T_{WI}			500	ns	
DRQ Cycle Time	T_{MCY}	13			μs	
DRQ Delay Time from $\overline{DACK}$ ↓	T_{AM}			200	ns	
TC Width	T_{TC}	1				ΦCY
Reset Width	T_{RST}	14			μs	ΦCY
WCK Cycle Time	T_{CY}		2 or 4② 1 or 2		μs	MFM = 0 MFM = 1
WCK Active Time (High)	T_0	80	250	350	ns	
WCK Rise Time	T_r			20	ns	
WCK Fall Time	T_f			20	ns	
Pre-Shift Delay Time from WCK ↑	T_{CP}	20		100	ns	
WDA Delay Time from WCK ↑	T_{CD}	20		100	ns	
RDD Active Time (High)	T_{RDD}	40			ns	
Window Cycle Time	T_{WCY}		2.0 1.0		μs	MFM = 0 MFM = 1
Window Hold Time to/from RDD	T_{RDW} T_{WRD}	15			ns	
$US_{0,1}$ Hold Time to RW/SEEK ↑	T_{US}	12			μs	8 MHz Clock Period
SEEK/RW Hold Time to LOW CURRENT/DIRECTION ↑	T_{SD}	7			μs	
LOW CURRENT/DIRECTION Hold Time to FAULT RESET/STEP ↑	T_{DST}	5.0			μs	
$US_{0,1}$ Hold Time from FAULT RESET/STEP ↑	T_{STU}	1.0			μs	
STEP Active Time (High)	T_{STP}		5.0		μs	
LOW CURRENT/DIRECTION Hold Time from FAULT RESET/STEP ↓	T_{STD}	5.0			μs	
STEP Cycle Time	T_{SC}	33	③	③	μs	
FAULT RESET Active Time (High)	T_{FR}	8.0		10	μs	
Write Data Width	T_{WDD}	T_0-50			ns	
$US_{0,1}$ Hold Time After SEEK	T_{SU}	15			μs	
Seek Hold Time from DIR	T_{DS}	30			μs	
DIR Hold Time after STEP	T_{STD}	24			μs	
Delay from RESET to INT	T_{RSI}	1250		1350	μs	
Index Pulse Width	T_{IDX}	625			μs	
DRQ Delay from $\overline{RD}$ ↓	T_{MR}	800			ns	8 MHz Clock Period
DRQ Delay from $\overline{WR}$ ↓	T_{MW}	250			ns	
$\overline{WE}$ or $\overline{RD}$ Response Time from DRQ ↑	T_{MRW}			12	μs	

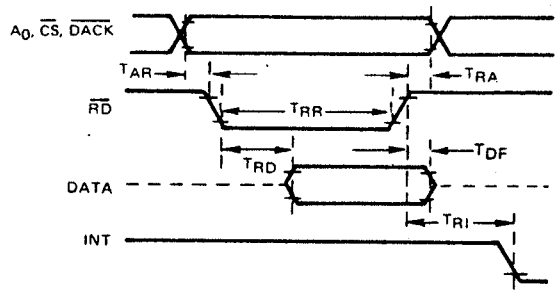
Notes: ① Typical values for T_a = 25°C and nominal supply voltage.

② The former value of 2 and 1 are applied to Standard Floppy, and the latter value of 4 and 2 are applied to Mini-floppy.

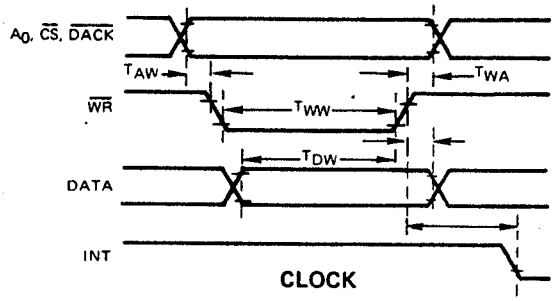
③ Under Software Control. The range is from 1 ms to 16 ms at 8 MHz Clock Period, and 2 to 32 ms at 4 MHz Clock Period.

TIMING WAVEFORMS

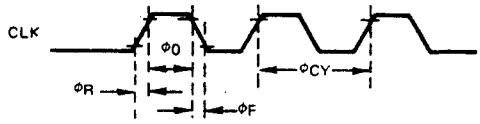
PROCESSOR READ OPERATION



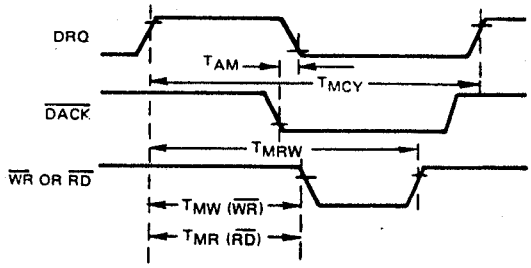
PROCESSOR WRITE OPERATION



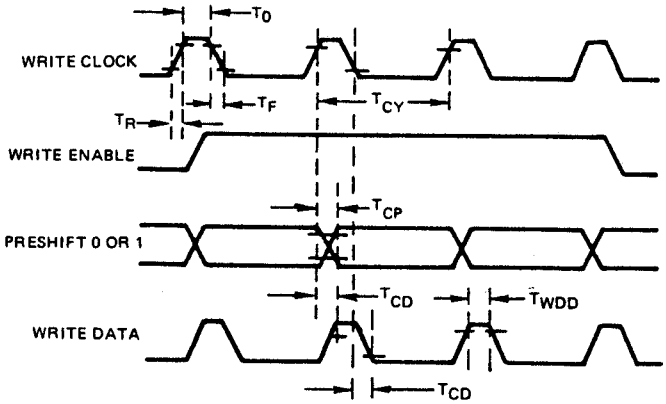
CLOCK



DMA OPERATION



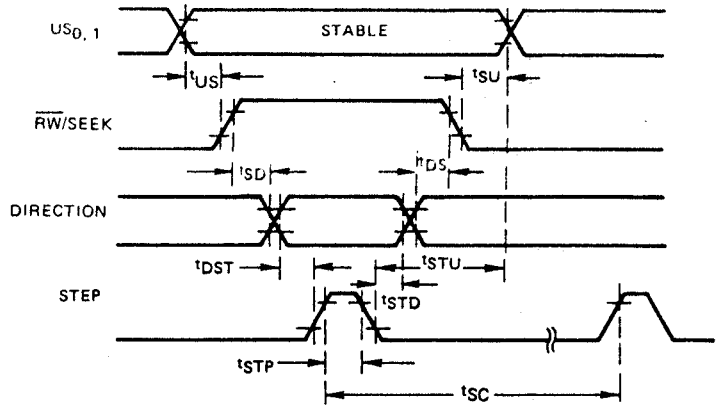
FDD WRITE OPERATION



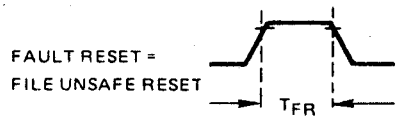
	PRESHIFT 0	PRESHIFT 1
NORMAL	0	0
LATE	0	1
EARLY	1	0
INVALID	1	1

TIMING WAVEFORMS (CONT.)

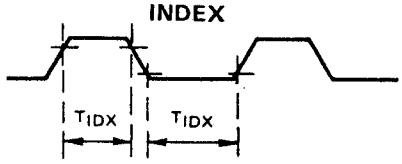
SEEK OPERATION



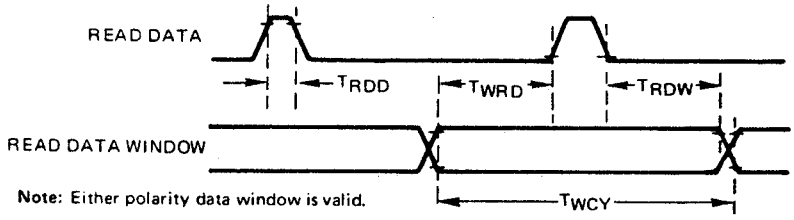
FLT RESET



INDEX

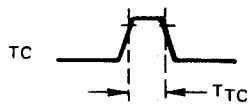


FDD READ OPERATION

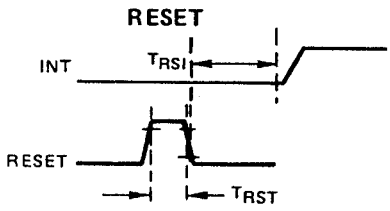


Note: Either polarity data window is valid.

TERMINAL COUNT



RESET



INTERNAL REGISTERS

The μPD765 contains two registers which may be accessed by the main system processor; a Status Register and a Data Register. The 8-bit Main Status Register contains the status information of the FDC, and may be accessed at any time. The 8-bit Data Register (actually consists of several registers in a stack with only one register presented to the data bus at a time), which stores data, commands, parameters, and FDD status information. Data bytes are read out of, or written into, the Data Register in order to program or obtain the results after a particular command. The Status Register may only be read and is used to facilitate the transfer of data between the processor and μPD765.

The relationship between the Status/Data registers and the signals $\overline{RD}$, $\overline{WR}$, and A_0 is shown below.

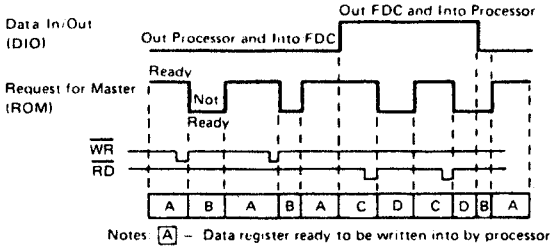
A_0	$\overline{RD}$	$\overline{WR}$	FUNCTION
0	0	1	Read Main Status Register
0	1	0	Illegal
0	0	0	Illegal
1	0	0	Illegal
1	0	1	Read from Data Register
1	1	0	Write into Data Register

INTERNAL REGISTERS (CONT.)

The bits in the Main Status Register are defined as follows:

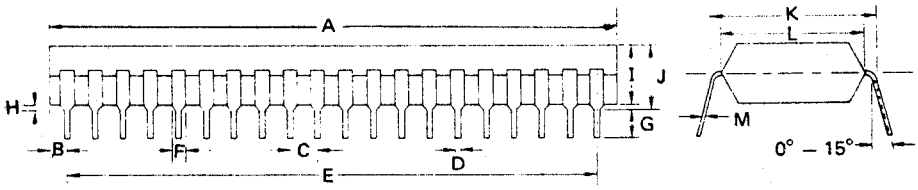
BIT NUMBER	NAME	SYMBOL	DESCRIPTION
DB_0	FDD 0 Busy	D_0B	FDD number 0 is in the Seek mode.
DB_1	FDD 1 Busy	D_1B	FDD number 1 is in the Seek mode.
DB_2	FDD 2 Busy	D_2B	FDD number 2 is in the Seek mode.
DB_3	FDD 3 Busy	D_3B	FDD number 3 is in the Seek mode.
DB_4	FDC Busy	CB	A read or write command is in process.
DB_5	Non-DMA mode	NDM	Indicates the FDC is in the non-DMA mode. This bit is set only during execution phase in non-DMA mode. When DB_5 goes low, execution phase has ended.
DB_6	Data Input/Output	DIO	Indicates direction of data transfer between FDC and Data Register. If DIO = "1" then transfer is from Data Register to the Processor. If DIO = "0", then transfer is from the Processor to Data Register.
DB_7	Request for Master	RQM	Indicates Data Register is ready to send or receive data to or from the Processor. Both bits DIO and RQM should be used to perform the hand-shaking functions of "ready" and "direction" to the processor.

The DIO and RQM bits in the Status Register indicate when Data is ready and in which direction data will be transferred on the Data Bus. The max time from the trailing edge of the last $\overline{RD}$ in the result phase to when DB_4 (FDC Busy) goes low is 12 μs.



Notes: A – Data register ready to be written into by processor
B – Data register not ready to be written into by processor
C – Data register ready for next data byte to be read by the processor
D – Data register not ready for next data byte to be read by processor

PACKAGE OUTLINE μPD765C



ITEM	MILLIMETERS	INCHES
A	51.5 MAX	2.028 MAX
B	1.62	0.064
C	2.54 ± 0.1	0.10 ± 0.004
D	0.5 ± 0.1	0.019 ± 0.004
E	48.26	1.9
F	1.2 MIN	0.047 MIN
G	2.54 MIN	0.10 MIN
H	0.5 MIN	0.019 MIN
I	5.22 MAX	0.206 MAX
J	5.72 MAX	0.225 MAX
K	15.24	0.600
L	13.2	0.520
M	$0.25^{+0.1}_{-0.05}$	$0.010^{+0.004}_{-0.002}$

COMMAND SEQUENCE

The μPD765 is capable of performing 15 different commands. Each command is initiated by a multi-byte transfer from the processor, and the result after execution of the command may also be a multi-byte transfer back to the processor. Because of this multi-byte interchange of information between the μPD765 and the processor, it is convenient to consider each command as consisting of three phases:

Command Phase: The FDC receives all information required to perform a particular operation from the processor.

Execution Phase: The FDC performs the operation it was instructed to do.

Result Phase: After completion of the operation, status and other housekeeping information are made available to the processor.

PHASE	R/W	DATA BUS D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	REMARKS
READ DATA										
Command	W	MT	MF	SK	0	0	1	1	0	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	C								Sector ID information prior to Command execution
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	DTL								
Execution										Data-transfer between the FDD and main-system
Result	R	ST 0								Status information after Command execution
	R	ST 1								
	R	ST 2								
	R	C								Sector ID information after Command execution
	R	H								
	R	R								
	R	N								
READ DELETED DATA										
Command	W	MT	MF	SK	0	1	1	0	0	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	C								Sector ID information prior to Command execution
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	DTL								
Execution										Data-transfer between the FDD and main-system
Result	R	ST 0								Status information after Command execution
	R	ST 1								
	R	ST 2								
	R	C								Sector ID information after Command execution
	R	H								
	R	R								
	R	N								
WRITE DATA										
Command	W	MT	MF	0	0	0	1	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	C								Sector ID information prior to Command execution
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	DTL								
Execution										Data-transfer between the main-system and FDD
Result	R	ST 0								Status information after Command execution
	R	ST 1								
	R	ST 2								
	R	C								Sector ID information after Command execution
	R	H								
	R	R								
	R	N								
WRITE DELETED DATA										
Command	W	MT	MF	0	0	1	0	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	C								Sector ID information prior to Command execution
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	DTL								
Execution										Data-transfer between the FDD and main-system
Result	R	ST 0								Status information after Command execution
	R	ST 1								
	R	ST 2								
	R	C								Sector ID information after Command execution
	R	H								
	R	R								
	R	N								

PHASE	R/W	DATA BUS D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	REMARKS
READ A TRACK										
Command	W	0	MF	SK	0	0	0	1	0	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	C								Sector ID information prior to Command execution
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	DTL								
Execution										Data-transfer between the FDD and main-system. FDC reads all data fields from index hole to EOT.
Result	R	ST 0								Status information after Command execution
	R	ST 1								
	R	ST 2								
	R	C								Sector ID information after Command execution
	R	H								
	R	R								
	R	N								
READ ID										
Command	W	0	MF	0	0	1	0	1	0	Commands
	W	X	X	X	X	X	HD	US1	US0	
Execution										The first correct ID information on the Cylinder is stored in Data Register
Result	R	ST 0								Status information after Command execution
	R	ST 1								
	R	ST 2								
	R	C								Sector ID information during Execution Phase
	R	H								
	R	R								
	R	N								
FORMAT A TRACK										
Command	W	0	MF	0	0	1	1	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	N								Bytes/Sector
	W	SC								Sectors/Track
	W	GPL								Gap 3
	W	D								Filler Byte
Execution										FDC formats an entire track
Result	R	ST 0								Status information after Command execution
	R	ST 1								
	R	ST 2								
	R	C								In this case, the ID information has no meaning
	R	H								
	R	R								
	R	N								
SCAN EQUAL										
Command	W	MT	MF	SK	1	0	0	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	C								Sector ID information prior to Command execution
	W	H								
	W	R								
	W	N								
	W	EOT								
	W	GPL								
	W	STP								
Execution										Data-compared between the FDD and main-system
Result	R	ST 0								Status information after Command execution
	R	ST 1								
	R	ST 2								
	R	C								Sector ID information after Command execution
	R	H								
	R	R								
	R	N								

Note: ① Symbols used in this table are described at the end of this section.

② A_0 should equal binary 1 for all operations.

③ X = Don't care, usually made to equal binary 0.

PHASE	R/W	DATA BUS D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	REMARKS
SCAN LOW OR EQUAL										
Command	W	MT	MF	SK	1	1	0	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	———			C				———	Sector ID information prior Command execution
	W	———			H				———	
	W	———			R				———	
	W	———			N				———	
	W	———			EOT				———	
	W	———			GPL				———	
	W	———			STP				———	
Execution										Data-compared between the FDD and main-system
Result	R	———			ST 0				———	Status information after Command execution
	R	———			ST 1				———	
	R	———			ST 2				———	
	R	———			C				———	Sector ID information after Command execution
	R	———			H				———	
	R	———			R				———	
	R	———			N				———	
SCAN HIGH OR EQUAL										
Command	W	MT	MF	SK	1	1	1	0	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	———			C				———	Sector ID information prior Command execution
	W	———			H				———	
	W	———			R				———	
	W	———			N				———	
	W	———			EOT				———	
	W	———			GPL				———	
	W	———			STP				———	
Execution										Data-compared between the FDD and main-system
Result	R	———			ST 0				———	Status information after Command execution
	R	———			ST 1				———	
	R	———			ST 2				———	
	R	———			C				———	Sector ID information after Command execution
	R	———			H				———	
	R	———			R				———	
	R	———			N				———	

PHASE	R/W	DATA BUS D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	REMARKS
RECALIBRATE										
Command	W	0	0	0	0	0	1	1	1	Command Codes
	W	X	X	X	X	X	0	US1	US0	
Execution										Head retracted to Track 0
SENSE INTERRUPT STATUS										
Command	W	0	0	0	0	1	0	0	0	Command Codes
Result	R	———			STO				———	Status information at the end of seek-operation about the FDC
	R	———			PCN				———	
SPECIFY										
Command	W	0	0	0	0	0	0	1	1	Command Codes
	W	—SRT							HUT—	
	W	—HLT							ND	
SENSE DRIVE STATUS										
Command	W	0	0	0	0	0	1	0	0	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
Result	R	———			ST 3				———	Status information about FDD
SEEK										
Command	W	0	0	0	0	1	1	1	1	Command Codes
	W	X	X	X	X	X	HD	US1	US0	
	W	———			NCN				———	
Execution										Head is positioned over proper Cylinder on Diskette
INVALID										
Command	W	———			Invalid Codes				———	Invalid Command Codes (NoOp – FDC goes into Standby State)
Result	R	———			ST 0				———	ST 0 = 80 (16)

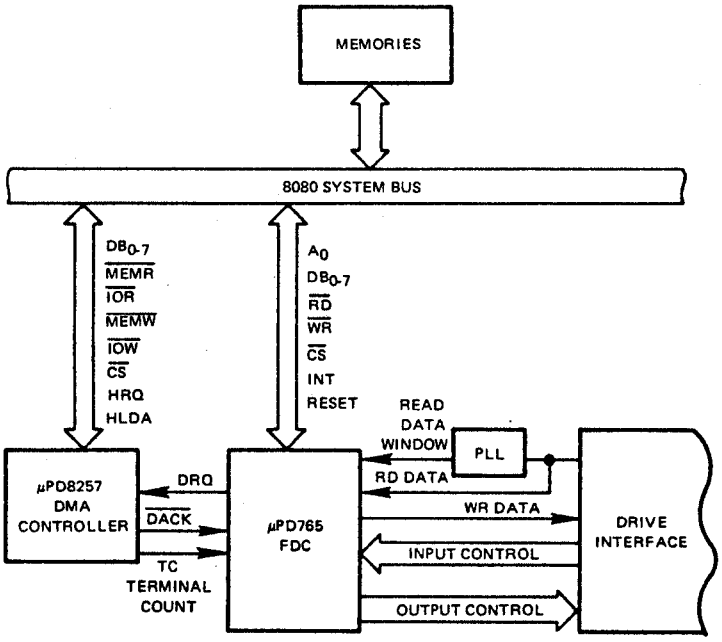
COMMAND SYMBOL DESCRIPTION

SYMBOL	NAME	DESCRIPTION
A_0	Address Line 0	A_0 controls selection of Main Status Register (A_0 = 0) or Data Register (A_0 = 1)
C	Cylinder Number	C stands for the current/selected Cylinder (track) number 0 through 76 of the medium.
D	Data	D stands for the data pattern which is going to be written into a Sector.
D_7-D_0	Data Bus	8-bit Data Bus, where D_7 stands for a most significant bit, and D_0 stands for a least significant bit.
DTL	Data Length	When N is defined as 00, DTL stands for the data length which users are going to read out or write into the Sector.
EOT	End of Track	EOT stands for the final Sector number on a Cylinder.
GPL	Gap Length	GPL stands for the length of Gap 3 (spacing between Sectors excluding VCO Sync. Field).
H	Head Address	H stands for head number 0 or 1, as specified in ID field.
HD	Head	HD stands for a selected head number 0 or 1. (H = HD in all command words.)
HLT	Head Load Time	HLT stands for the head load time in the FDD (2 to 254 ms in 2 ms increments).
HUT	Head Unload Time	HUT stands for the head unload time after a read or write operation has occurred (16 to 240 ms in 16 ms increments).
MF	FM or MFM Mode	If MF is low, FM mode is selected, and if it is high, MFM mode is selected.
MT	Multi-Track	If MT is high, a multi-track operation is to be performed. (A cylinder under both HD0 and HD1 will be read or written.)

COMMAND SYMBOL DESCRIPTION (CONT.)

SYMBOL	NAME	DESCRIPTION
N	Number	N stands for the number of data bytes written in a Sector.
NCN	New Cylinder Number	NCN stands for a new Cylinder number, which is going to be reached as a result of the Seek operation. Desired position of Head.
ND	Non-DMA Mode	ND stands for operation in the Non-DMA Mode.
PCN	Present Cylinder Number	PCN stands for the Cylinder number at the completion of SENSE INTERRUPT STATUS Command. Position of Head at present time.
R	Record	R stands for the Sector number, which will be read or written.
R/W	Read/Write	R/W stands for either Read (R) or Write (W) signal.
SC	Sector	SC indicates the number of Sectors per Cylinder.
SK	Skip	SK stands for Skip Deleted Data Address Mark.
SRT	Step Rate Time	SRT stands for the Stepping Rate for the FDD. (1 to 16 ms in 1 ms increments.) Stepping Rate applies to all drives, (F = 1 ms, E = 2 ms, etc.).
ST 0 ST 1 ST 2 ST 3	Status 0 Status 1 Status 2 Status 3	ST 0-3 stand for one of four registers which store the status information after a command has been executed. This information is available during the result phase after command execution. These registers should not be confused with the main status register (selected by $A_0 = 0$). ST 0-3 may be read only after a command has been executed and contain information relevant to that particular command.
STP		During a Scan operation, if STP = 1, the data in contiguous sectors is compared byte by byte with data sent from the processor (or DMA); and if STP = 2, then alternate sectors are read and compared.
US0, US1	Unit Select	US stands for a selected drive number 0 or 1.

SYSTEM CONFIGURATION



PROCESSOR INTERFACE

During Command or Result Phases the Main Status Register (described earlier) must be read by the processor before each byte of information is written into or read from the Data Register. Bits D6 and D7 in the Main Status Register must be in a 0 and 1 state, respectively, before each byte of the command word may be written into the μPD765. Many of the commands require multiple bytes, and as a result the Main Status Register must be read prior to each byte transfer to the μPD765. On the other hand, during the Result Phase, D6 and D7 in the Main Status Register must both be 1's (D6 = 1 and D7 = 1) before reading each byte from the Data Register. Note, this reading of the Main Status Register before each byte transfer to the μPD765 is required in only the Command and Result Phases, and NOT during the Execution Phase.

During the Execution Phase, the Main Status Register need not be read. If the μPD765 is in the NON-DMA Mode, then the receipt of each data byte (if μPD765 is reading data from FDD) is indicated by an Interrupt signal on pin 18 (INT = 1). The generation of a Read signal (RD = 0) will reset the Interrupt as well as output the Data onto the Data Bus. If the processor cannot handle Interrupts fast enough (every 13 μs) then it may poll the Main Status Register and then bit D7 (RQM) functions just like the Interrupt signal. If a Write Command is in process then the WR signal performs the reset to the Interrupt signal.

If the μPD765 is in the DMA Mode, no Interrupts are generated during the Execution Phase. The μPD765 generates DRQ's (DMA Requests) when each byte of data is available. The DMA Controller responds to this request with both a DACK = 0 (DMA Acknowledge) and a RD = 0 (Read signal). When the DMA Acknowledge signal goes low (DACK = 0) then the DMA Request is reset (DRQ = 0). If a Write Command has been programmed then a WR signal will appear instead of RD. After the Execution Phase has been completed (Terminal Count has occurred) then an Interrupt will occur (INT = 1). This signifies the beginning of the Result Phase. When the first byte of data is read during the Result Phase, the Interrupt is automatically reset (INT = 0).

It is important to note that during the Result Phase all bytes shown in the Command Table must be read. The Read Data Command, for example has seven bytes of data in the Result Phase. All seven bytes must be read in order to successfully complete the Read Data Command. The μPD765 will not accept a new command until all seven bytes have been read. Other commands may require fewer bytes to be read during the Result Phase.

The μPD765 contains five Status Registers. The Main Status Register mentioned above may be read by the processor at any time. The other four Status Registers (ST0, ST1, ST2, and ST3) are only available during the Result Phase, and may be read only after successfully completing a command. The particular command which has been executed determines how many of the Status Registers will be read.

The bytes of data which are sent to the μPD765 to form the Command Phase, and are read out of the μPD765 in the Result Phase, must occur in the order shown in the Command Table. That is, the Command Code must be sent first and the other bytes sent in the prescribed sequence. No foreshortening of the Command or Result Phases are allowed. After the last byte of data in the Command Phase is sent to the μPD765, the Execution Phase automatically starts. In a similar fashion, when the last byte of data is read out in the Result Phase, the command is automatically ended and the μPD765 is ready for a new command. A command may be truncated (prematurely ended) by simply sending a Terminal Count signal to pin 16 (TC = 1). This is a convenient means of ensuring that the processor may always get the μPD765's attention even if the disk system hangs up in an abnormal manner.

POLLING FEATURE OF THE μPD765

After the Specify command has been sent to the μPD765, the Unit Select line US0 and US1 will automatically go into a polling mode. In between commands (and between step pulses in the SEEK command) the μPD765 polls all four FDD's looking for a change in the Ready line from any of the drives. If the Ready line changes state (usually due to a door opening or closing) then the μPD765 will generate an interrupt. When Status Register 0 (STO) is read (after Sense Interrupt Status is issued), Not Ready (NR) will be indicated. The polling of the Ready line by the μPD765 occurs continuously between instructions, thus notifying the processor which drives are on or off line.

READ DATA

A set of nine (9) byte words are required to place the FDC into the Read Data Mode. After the Read Data command has been issued the FDC loads the head (if it is in the unloaded state), waits the specified head settling time (defined in the Specify Command), and begins reading ID Address Marks and ID fields. When the current sector number ("R") stored in the ID Register (IDR) compares with the sector number read off the diskette, then the FDC outputs data (from the data field) byte-to-byte to the main system via the data bus.

After completion of the read operation from the current sector, the Sector Number is incremented by one, and the data from the next sector is read and output on the data bus. This continuous read function is called a "Multi-Sector Read Operation." The Read Data Command may be terminated by the receipt of a Terminal Count signal. Upon receipt of this signal, the FDC stops outputting data to the processor, but will continue to read data from the current sector, check CRC (Cyclic Redundancy Count) bytes, and then at the end of the sector terminate the Read Data Command.

The amount of data which can be handled with a single command to the FDC depends upon MT (multi-track), MF (MFM/FM), and N (Number of Bytes/Sector). Table 1 below shows the Transfer Capacity.

Multi-Track MT	MFM/FM MF	Bytes/Sector N	Maximum Transfer Capacity (Bytes/Sector) (Number of Sectors)	Final Sector Read from Diskette
0	0	00	(128) (26) = 3,328	26 at Side 0
0	1	01	(256) (26) = 6,656	or 26 at Side 1
1	0	00	(128) (52) = 6,656	26 at Side 1
1	1	01	(256) (52) = 13,312	
0	0	01	(256) (15) = 3,840	15 at Side 0
0	1	02	(512) (15) = 7,680	or 15 at Side 1
1	0	01	(256) (30) = 7,680	15 at Side 1
1	1	02	(512) (30) = 15,360	
0	0	02	(512) (8) = 4,096	8 at Side 0
0	1	03	(1024) (8) = 8,192	or 8 at Side 1
1	0	02	(512) (16) = 8,192	8 at Side 1
1	1	03	(1024) (16) = 16,384	

Table 1. Transfer Capacity

The "multi-track" function (MT) allows the FDC to read data from both sides of the diskette. For a particular cylinder, data will be transferred starting at Sector 0, Side 0 and completing at Sector L, Side 1 (Sector L = last sector on the side). Note, this function pertains to only one cylinder (the same track) on each side of the diskette.

When N = 0, then DTL defines the data length which the FDC must treat as a sector. If DTL is smaller than the actual data length in a Sector, the data beyond DTL in the Sector, is not sent to the Data Bus. The FDC reads (internally) the complete Sector performing the CRC check, and depending upon the manner of command termination, may perform a Multi-Sector Read Operation. When N is non-zero, then DTL has no meaning and should be set to FF Hexidecimal.

At the completion of the Read Data Command, the head is not unloaded until after Head Unload Time Interval (specified in the Specify Command) has elapsed. If the processor issues another command before the head unloads then the head settling time may be saved between subsequent reads. This time out is particularly valuable when a diskette is copied from one drive to another.

If the FDC detects the Index Hole twice without finding the right sector, (indicated in "R"), then the FDC sets the ND (No Data) flag in Status Register 1 to a 1 (high), and terminates the Read Data Command. (Status Register 0 also has bits 7 and 6 set to 0 and 1 respectively.)

After reading the ID and Data Fields in each sector, the FDC checks the CRC bytes. If a read error is detected (incorrect CRC in ID field), the FDC sets the DE (Data Error) flag in Status Register 1 to a 1 (high), and if a CRC error occurs in the Data Field the FDC also sets the DD (Data Error in Data Field) flag in Status Register 2 to a 1 (high), and terminates the Read Data Command. (Status Register 0 also has bits 7 and 6 set to 0 and 1 respectively.)

Status Register 2 to a 1 (high), and terminates the Read Data Command.

If the FDC reads a Deleted Data Address Mark off the diskette, and the SK bit (bit D5 in the first Command Word) is not set (SK = 0), then the FDC sets the CM (Control Mark) flag in Status Register 2 to a 1 (high), and terminates the Read Data Command, after reading all the data in the Sector. If SK = 1, the FDC skips the sector with the Deleted Data Address Mark and reads the next sector. The CRC bits in the deleted data field are not checked when SK = 1.

During disk data transfers between the FDC and the processor, via the data bus, the FDC must be serviced by the processor every 27 μs in the FM Mode, and every 13 μs in the MFM Mode, or the FDC sets the OR (Over Run) flag in Status Register 1 to a 1 (high), and terminates the Read Data Command.

If the processor terminates a read (or write) operation in the FDC, then the ID Information in the Result Phase is dependent upon the state of the MT bit and EOT byte. Table 2 shows the values for C, H, R, and N, when the processor terminates the Command.

FUNCTIONAL DESCRIPTION OF COMMANDS (CONT.)

MT	EOT	Final Sector Transferred to Processor	ID Information at Result Phase			
			C	H	R	N
0	1A	Sector 1 to 25 at Side 0	NC	NC	R + 1	NC
	0F	Sector 1 to 14 at Side 0				
	08	Sector 1 to 7 at Side 0				
	1A	Sector 26 at Side 0	C + 1	NC	R = 01	NC
	0F	Sector 15 at Side 0				
	08	Sector 8 at Side 0				
	1A	Sector 1 to 25 at Side 1	NC	NC	R + 1	NC
	0F	Sector 1 to 14 at Side 1				
	08	Sector 1 to 7 at Side 1				
	1A	Sector 26 at Side 1	C + 1	NC	R = 01	NC
	0F	Sector 15 at Side 1				
	08	Sector 8 at Side 1				
1	1A	Sector 1 to 25 at Side 0	NC	NC	R + 1	NC
	0F	Sector 1 to 14 at Side 0				
	08	Sector 1 to 7 at Side 0				
	1A	Sector 26 at Side 0	NC	LSB	R = 01	NC
	0F	Sector 15 at Side 0				
	08	Sector 8 at Side 0				
	1A	Sector 1 to 25 at Side 1	NC	NC	R + 1	NC
	0F	Sector 1 to 14 at Side 1				
	08	Sector 1 to 7 at Side 1				
	1A	Sector 26 at Side 1	C + 1	LSB	R = 01	NC
	0F	Sector 15 at Side 1				
	08	Sector 8 at Side 1				

Notes: 1 NC (No Change): The same value as the one at the beginning of command execution.

2 LSB (Least Significant Bit): The least significant bit of H is complemented.

Table 2: ID Information When Processor Terminates Command

WRITE DATA

A set of nine (9) bytes are required to set the FDC into the Write Data mode. After the Write Data command has been issued the FDC loads the head (if it is in the unloaded state), waits the specified heat settling time (defined in the Specify Command), and begins reading ID Fields. When the current sector number ("R"), stored in the ID Register (IDR) compares with the sector number read off the diskette, then the FDC takes data from the processor byte-by-byte via the data bus, and outputs it to the FDD.

After writing data into the current sector, the Sector Number stored in "R" is incremented by one, and the next data field is written into. The FDC continues this "Multi-Sector Write Operation" until the issuance of a Terminal Count signal. If a Terminal Count signal is sent to the FDC it continues writing into the current sector to complete the data field. If the Terminal Count signal is received while a data field is being written then the remainder of the data field is filled with 00 (zeros).

The FDC reads the ID field of each sector and checks the CRC bytes. If the FDC detects a read error (incorrect CRC) in one of the ID Fields, it sets the DE (Data Error) flag of Status Register 1 to a 1 (high), and terminates the Write Data Command. (Status Register 0 also has bits 7 and 6 set to 0 and 1 respectively.)

The Write Command operates in much the same manner as the Read Command. The following items are the same, and one should refer to the Read Data Command for details:

- Transfer Capacity
- EN (End of Cylinder) Flag
- ND (No Data) Flag
- Head Unload Time Interval
- ID Information when the processor terminates command (see Table 2)
- Definition of DTL when N = 0 and when N ≠ 0

In the Write Data mode, data transfers between the processor and FDC, via the Data Bus, must occur every 31 μs in the FM mode, and every 15 μs in the MFM mode. If the time interval between data transfers is longer than this then the FDC sets the OR (Over Run) flag in Status Register 1 to a 1 (high), and terminates the Write Data Command. (Status Register 0 also has bit 7 and 6 set to 0 and 1 respectively.)

WRITE DELETED DATA

This command is the same as the Write Data Command except a Deleted Data Address Mark is written at the beginning of the Data Field instead of the normal Data Address Mark.

READ DELETED DATA

This command is the same as the Read Data Command except that when the FDC detects a Data Address Mark at the beginning of a Data Field (and SK = 0 (low), it will read all the data in the sector and set the CM flag in Status Register 2 to a 1 (high), and then terminate the command. If SK = 1, then the FDC skips the sector with the Data Address Mark and reads the next sector.

READ A TRACK

This command is similar to READ DATA Command except that this is a continuous READ operation where the entire data field from each of the sectors are read. Immediately after encountering the INDEX HOLE, the FDC starts reading all data fields on the track, as continuous blocks of data. If the FDC finds an error in the ID or DATA CRC check bytes, it continues to read data from the track. The FDC compares the ID information read from each sector with the value stored in the IDR, and sets the ND flag of Status Register 1 to a 1 (high) if there is no comparison. Multi-track or skip operations are not allowed with this command.

This command terminates when EOT number of sectors have been read. If the FDC does not find an ID Address Mark on the diskette after it encounters the INDEX HOLE for the second time, then it sets the MA (missing address mark) flag in Status Register 1 to a 1 (high), and terminates the command. (Status Register 0 has bits 7 and 6 set to 0 and 1 respectively.)

READ ID

The READ ID Command is used to give the present position of the recording head. The FDC stores the values from the first ID Field it is able to read. If no proper ID Address Mark is found on the diskette, before the INDEX HOLE is encountered for the second time then the MA (Missing Address Mark) flag in Status Register 1 is set to a 1 (high), and if no data is found then the ND (No Data) flag is also set in Status Register 1 to a 1 (high). The command is then terminated with Bits 7 and 6 in Status Register 0 set to 0 and 1 respectively.

FORMAT A TRACK

The Format Command allows an entire track to be formatted. After the INDEX HOLE is detected, Data is written on the Diskette; Gaps, Address Marks, ID Fields and Data Fields, all per the IBM System 34 (Double Density) or System 3740 (Single Density) Format are recorded. The particular format which will be written is controlled by the values programmed into N (number of bytes/sector), SC (sectors/cylinder), GPL (Gap Length), and D (Data Pattern) which are supplied by the processor during the Command Phase. The Data Field is filled with the Byte of data stored in D. The ID Field for each sector is supplied by the processor; that is, four data requests per sector are made by the FDC for C (Cylinder Number), H (Head Number), R (Sector Number) and N (Number of Bytes/Sector). This allows the diskette to be formatted with non-sequential sector numbers, if desired.

After formatting each sector, the processor must send new values for C, H, R, and N to the μPD765 for each sector on the track. The contents of the R register is incremented by one after each sector is formatted, thus, the R register contains a value of R when it is read during the Result Phase. This incrementing and formatting continues for the whole track until the FDC encounters the INDEX HOLE for the second time, whereupon it terminates the command.

If a FAULT signal is received from the FDD at the end of a write operation, then the FDC sets the EC flag of Status Register 0 to a 1 (high), and terminates the command after setting bits 7 and 6 of Status Register 0 to 0 and 1 respectively. Also the loss of a READY signal at the beginning of a command execution phase causes bits 7 and 6 of Status Register 0 to be set to 0 and 1 respecitvely.

Table 3 shows the relationship between N, SC, and GPL for various sector sizes:

FORMAT	SECTOR SIZE	N	SC	GPL ①	GPL ②	REMARKS
FM Mode	128 bytes/Sector	00	$1A_{(16)}$	$.07_{(16)}$	$1B_{(16)}$	IBM Diskette 1
	256	01	$0F_{(16)}$	$0E_{(16)}$	$2A_{(16)}$	IBM Diskette 2
	512	02	08	$1B_{(16)}$	$3A_{(16)}$	
FM Mode	1024 bytes/Sector	03	04	–	–	
	2048	04	02	–	–	
	4096	05	01	–	–	
MFM Mode	256	01	$1A_{(16)}$	$0E_{(16)}$	$36_{(16)}$	IBM Diskette 2D
	512	02	$0F_{(16)}$	$1B_{(16)}$	$54_{(16)}$	
	1024	03	08	$35_{(16)}$	$74_{(16)}$	IBM Diskette 2D
	2048	04	04	–	–	
	4096	05	02	–	–	
	8192	06	01	–	–	

Table 3

Note: ① Suggested values of GPL in Read or Write Commands to avoid splice point between data field and ID field of contiguous sections.

② Suggested values of GPL in format command.

FUNCTIONAL DESCRIPTION OF COMMANDS (CONT.)

SCAN COMMANDS

The SCAN Commands allow data which is being read from the diskette to be compared against data which is being supplied from the main system (Processor in NON-DMA mode, and DMA Controller in DMA mode). The FDC compares the data on a byte-by-byte basis, and looks for a sector of data which meets the conditions of $D_{FDD} = D_{Processor}$, $D_{FDD} \leq D_{Processor}$, or $D_{FDD} \geq D_{Processor}$. Ones complement arithmetic is used for comparison (FF = largest number, 00 = smallest number). After a whole sector of data is compared, if the conditions are not met, the sector number is incremented (R + STP → R), and the scan operation is continued. The scan operation continues until one of the following conditions occur; the conditions for scan are met (equal, low, or high), the last sector on the track is reached (EOT), or the terminal count signal is received.

If the conditions for scan are met then the FDC sets the SH (Scan Hit) flag of Status Register 2 to a 1 (high), and terminates the Scan Command. If the conditions for scan are not met between the starting sector (as specified by R) and the last sector on the cylinder (EOT), then the FDC sets the SN (Scan Not Satisfied) flag of Status Register 2 to a 1 (high), and terminates the Scan Command. The receipt of a TERMINAL COUNT signal from the Processor or DMA Controller during the scan operation will cause the FDC to complete the comparison of the particular byte which is in process, and then to terminate the command. Table 4 shows the status of bits SH and SN under various conditions of SCAN.

COMMAND	STATUS REGISTER 2		COMMENTS
	BIT 2 = SN	BIT 3 = SH	
Scan Equal	0	1	$D_{FDD} = D_{Processor}$
	1	0	$D_{FDD} \neq D_{Processor}$
Scan Low or Equal	0	1	$D_{FDD} = D_{Processor}$
	0	0	$D_{FDD} < D_{Processor}$
	1	0	$D_{FDD} \not< D_{Processor}$
Scan High or Equal	0	1	$D_{FDD} = D_{Processor}$
	0	0	$D_{FDD} < D_{Processor}$
	1	0	$D_{FDD} \not> D_{Processor}$

Table 4

If the FDC encounters a Deleted Data Address Mark on one of the sectors (and SK = 0), then it regards the sector as the last sector on the cylinder, sets CM (Control Mark) flag of Status Register 2 to a 1 (high) and terminates the command. If SK = 1, the FDC skips the sector with the Deleted Address Mark, and reads the next sector. In the second case (SK = 1), the FDC sets the CM (Control Mark) flag of Status Register 2 to a 1 (high) in order to show that a Deleted Sector had been encountered.

When either the STP (contiguous sectors = 01, or alternate sectors = 02 sectors are read) or the MT (Multi-Track) are programmed, it is necessary to remember that the last sector on the track must be read. For example, if STP = 02, MT = 0, the sectors are numbered sequentially 1 through 26, and we start the Scan Command at sector 21; the following will happen. Sectors 21, 23, and 25 will be read, then the next sector (26) will be skipped and the Index Hole will be encountered before the EOT value of 26 can be read. This will result in an abnormal termination of the command. If the EOT had been set at 25 or the scanning started at sector 20, then the Scan Command would be completed in a normal manner.

During the Scan Command data is supplied by either the processor or DMA Controller for comparison against the data read from the diskette. In order to avoid having the OR (Over Run) flag set in Status Register 1, it is necessary to have the data available in less than 27 μs (FM Mode) or 13 μs (MFM Mode). If an Overrun occurs the FDC ends the command with bits 7 and 6 of Status Register 0 set to 0 and 1, respectively.

SEEK

The read/write head within the FDD is moved from cylinder to cylinder under control of the Seek Command. The FDC compares the PCN (Present Cylinder Number) which is the current head position with the NCN (New Cylinder Number), and if there is a difference performs the following operation:

PCN < NCN: Direction signal to FDD set to a 1 (high), and Step Pulses are issued. (Step In.)
PCN > NCN: Direction signal to FDD set to a 0 (low), and Step Pulses are issued. (Step Out.)

The rate at which Step Pulses are issued is controlled by SRT (Stepping Rate Time) in the SPECIFY Command. After each Step Pulse is issued NCN is compared against PCN, and when NCN = PCN, then the SE (Seek End) flag is set in Status Register 0 to a 1 (high), and the command is terminated.

During the Command Phase of the Seek operation the FDC is in the FDC BUSY state, but during the Execution Phase it is in the NON BUSY state. While the FDC is in the NON BUSY state, another Seek Command may be issued, and in this manner parallel seek operations may be done on up to 4 Drives at once.

If an FDD is in a NOT READY state at the beginning of the command execution phase or during the seek operation, then the NR (NOT READY) flag is set in Status Register 0 to a 1 (high), and the command is terminated after bits 7 and 6 of Status Register 0 are set to 0 and 1 respectively.

RECALIBRATE

The function of this command is to retract the read/write head within the FDD to the Track 0 position. The FDC clears the contents of the PCN counter, and checks the status of the Track 0 signal from the FDD. As long as the Track 0 signal is low, the Direction signal remains 1 (high) and Step Pulses are issued. When the Track 0 signal goes high, the SE (SEEK END) flag in Status Register 0 is set to a 1 (high) and the command is terminated. If the Track 0 signal is still low after 77 Step Pulse have been issued, the FDC sets the SE (SEEK END) and EC (EQUIPMENT CHECK) flags of Status Register 0 to both 1s (highs), and terminates the command after bits 7 and 6 of Status Register 0 is set to 0 and 1 respectively.

The ability to do overlap RECALIBRATE Commands to multiple FDDs and the loss of the READY signal, as described in the SEEK Command, also applies to the RECALIBRATE Command.

SENSE INTERRUPT STATUS

An Interrupt signal is generated by the FDC for one of the following reasons:

1. Upon entering the Result Phase of:
   a. Read Data Command
   b. Read a Track Command
   c. Read ID Command
   d. Read Deleted Data Command
   e. Write Data Command
   f. Format a Cylinder Command
   g. Write Deleted Data Command
   h. Scan Commands
2. Ready Line of FDD changes state
3. End of Seek or Recalibrate Command
4. During Execution Phase in the NON-DMA Mode

Interrupts caused by reasons 1 and 4 above occur during normal command operations and are easily discernible by the processor. However, interrupts caused by reasons 2 and 3 above may be uniquely identified with the aid of the Sense Interrupt Status Command. This command when issued resets the interrupt signal and via bits 5, 6, and 7 of Status Register 0 identifies the cause of the interrupt.

SEEK END BIT 5	INTERRUPT CODE		CAUSE
	BIT 6	BIT 7	
0	1	1	Ready Line changed state, either polarity
1	0	0	Normal Termination of Seek or Recalibrate Command
1	1	0	Abnormal Termination of Seek or Recalibrate Command

Table 5

Neither the Seek or Recalibrate Command have a Result Phase. Therefore, it is mandatory to use the Sense Interrupt Status Command after these commands to effectively terminate them and to provide verification of where the head is positioned (PCN).

SPECIFY

The Specify Command sets the initial values for each of the three internal timers. The HUT (Head Unload Time) defines the time from the end of the Execution Phase of one of the Read/Write Commands to the head unload state. This timer is programmable from 16 to 240 ms in increments of 16 ms (01 = 16 ms, 02 = 32 ms . . . . OF = 240 ms). The SRT (Step Rate Time) defines the time interval between adjacent step pulses. This timer is programmable from 1 to 16 ms in increments of 1 ms (F = 1 ms, E = 2 ms, D = 3 ms, etc.). The HLT (Head Load Time) defines the time between when the Head Load signal goes high and when the Read/Write operation starts. This timer is programmable from 2 to 254 ms in increments of 2 ms (01 = 2 ms, 02 = 4 ms, 03 = 6 ms . . . FE = 254 ms).

The time intervals mentioned above are a direct function of the clock (CLK on pin 19). Times indicated above are for an 8 MHz clock, if the clock was reduced to 4 MHz (mini-floppy application) then all time intervals are increased by a factor of 2.

The choice of DMA or NON-DMA operation is made by the ND (NON-DMA) bit. When this bit is high (ND = 1) the NON-DMA mode is selected, and when ND = 0 the DMA mode is selected.

SENSE DRIVE STATUS

This command may be used by the processor whenever it wishes to obtain the status of the FDDs. Status Register 3 contains the Drive Status information.

INVALID

If an invalid command is sent to the FDC (a command not defined above), then the FDC will terminate the command after bits 7 and 6 of Status Register 0 are set to 1 and 0 respectively. No interrupt is generated by the μPD765 during this condition. Bit 6 and bit 7 (DIO and RQM) in the Main Status Register are both high ("1") indicating to the processor that the μPD765 is in the Result Phase and the contents of Status Register 0 (ST0) must be read. When the processor reads Status Register 0 it will find a 80 hex indicating an invalid command was received.

A Sense Interrupt Status Command must be sent after a Seek or Recalibrate Interrupt, otherwise the FDC will consider the next command to be an Invalid Command.

In some applications the user may wish to use this command as a No-Op command, to place the FDC in a standby or no operation state.

STATUS REGISTER IDENTIFICATION

BIT NO.	BIT NAME	BIT SYMBOL	DESCRIPTION
STATUS REGISTER 0			
D_7	Interrupt Code	IC	$D_7 = 0$ and $D_6 = 0$ Normal Termination of Command, (NT). Command was completed and properly executed.
D_6			$D_7 = 0$ and $D_6 = 1$ Abnormal Termination of Command, (AT). Execution of Command was started, but was not successfully completed.
			$D_7 = 1$ and $D_6 = 0$ Invalid Command issue, (IC). Command which was issued was never started.
			$D_7 = 1$ and $D_6 = 1$ Abnormal Termination because during command execution the ready signal from FDD changed state.
D_5	Seek End	SE	When the FDC completes the SEEK Command, this flag is set to 1 (high).
D_4	Equipment Check	EC	If a fault Signal is received from the FDD, or if the Track 0 Signal fails to occur after 77 Step Pulses (Recalibrate Command) then this flag is set.
D_3	Not Ready	NR	When the FDD is in the not-ready state and a read or write command is issued, this flag is set. If a read or write command is issued to Side 1 of a single sided drive, then this flag is set.
D_2	Head Address	HD	This flag is used to indicate the state of the head at Interrupt.
D_1	Unit Select 1	US 1	These flags are used to indicate a Drive Unit Number at Interrupt
D_0	Unit Select 0	US 0	
STATUS REGISTER 1			
D_7	End of Cylinder	EN	When the FDC tries to access a Sector beyond the final Sector of a Cylinder, this flag is set.
D_6			Not used. This bit is always 0 (low).
D_5	Data Error	DE	When the FDC detects a CRC error in either the ID field or the data field, this flag is set.
D_4	Over Run	OR	If the FDC is not serviced by the main-systems during data transfers, within a certain time interval, this flag is set.
D_3			Not used. This bit always 0 (low).
D_2	No Data	ND	During execution of READ DATA, WRITE DELETED DATA or SCAN Command, if the FDC cannot find the Sector specified in the IDR Register, this flag is set.
			During executing the READ ID Command, if the FDC cannot read the ID field without an error, then this flag is set.
			During the execution of the READ A Cylinder Command, if the starting sector cannot be found, then this flag is set.

STATUS REGISTER IDENTIFICATION (CONT.)

BIT NO.	BIT NAME	BIT SYMBOL	DESCRIPTION
STATUS REGISTER 1 (CONT.)			
D_1	Not Writable	NW	During execution of WRITE DATA, WRITE DELETED DATA or Format A Cylinder Command, if the FDC detects a write protect signal from the FDD, then this flag is set.
D_0	Missing Address Mark	MA	If the FDC cannot detect the ID Address Mark after encountering the index hole twice, then this flag is set.
			If the FDC cannot detect the Data Address Mark or Deleted Data Address Mark, this flag is set. Also at the same time, the MD (Missing Address Mark in Data Field) of Status Register 2 is set.
STATUS REGISTER 2			
D_7			Not used. This bit is always 0 (low).
D_6	Control Mark	CM	During executing the READ DATA or SCAN Command, if the FDC encounters a Sector which contains a Deleted Data Address Mark, this flag is set.
D_5	Data Error in Data Field	DD	If the FDC detects a CRC error in the data field then this flag is set.
D_4	Wrong Cylinder	WC	This bit is related with the ND bit, and when the contents of C on the medium is different from that stored in the IDR, this flag is set.
D_3	Scan Equal Hit	SH	During execution, the SCAN Command, if the condition of "equal" is satisfied, this flag is set.
D_2	Scan Not Satisfied	SN	During executing the SCAN Command, if the FDC cannot find a Sector on the cylinder which meets the condition, then this flag is set.
D_1	Bad Cylinder	BC	This bit is related with the ND bit, and when the content of C on the medium is different from that stored in the IDR and the content of C is FF, then this flag is set.
D_0	Missing Address Mark in Data Field	MD	When data is read from the medium, if the FDC cannot find a Data Address Mark or Deleted Data Address Mark, then this flag is set.
STATUS REGISTER 3			
D_7	Fault	FT	This bit is used to indicate the status of the Fault signal from the FDD.
D_6	Write Protected	WP	This bit is used to indicate the status of the Write Protected signal from the FDD.
D_5	Ready	RY	This bit is used to indicate the status of the Ready signal from the FDD.
D_4	Track 0	T0	This bit is used to indicate the status of the Track 0 signal from the FDD.
D_3	Two Side	TS	This bit is used to indicate the status of the Two Side signal from the FDD.
D_2	Head Address	HD	This bit is used to indicate the status of Side Select signal to the FDD.
D_1	Unit Select 1	US 1	This bit is used to indicate the status of the Unit Select 1 signal to the FDD.
D_0	Unit Select 0	US 0	This bit is used to indicate the status of the Unit Select 0 signal to the FDD.

ERRATA
Edition 2, FDC-2 Manual
July 10, 1980

The following material will be incorporated into Edition 3 of the FDC-2 Manual.

Disc Drive Power Supplies

Users should be aware that the FDC-2 requires that the disc drive power supply be capable of supporting all system stepper motors simultaneously. A typical requirement for a single drive would be 24 Volts at 1.4 Amps; a four drive system of such typical drives would require 24 Volts at 5.6 Amps. As the manual explains, the drives should be jumpered so that their stepper motors are always enabled and do not require drive select to be enabled.

The advantage of this arrangement is that the FDC-2 can perform simultaneous stepping by rapidly multiplexing drive select from one drive to the next; this rapid multiplexing of drive select, however, is not suitable for the requirements of a stepper motor that is enabled by drive select.

Disk Drive Setup

The user should be aware that user selectable options existing in various manufacturers' floppy disk drives must be correctly selected to insure dependable operation with the FDC2 controller.

Stepper Motor Enable:

For correct operation with the FDC2, the floppy disk drive should be jumpered to have a continously enabled stepper motor. The stepper motor should NOT require active drive select or head load in order to be enabled.
To select this mode of operation in:

Shugart 800/850, Remex 4000, Qume ; open Jumpers HL and DS

Head Load:

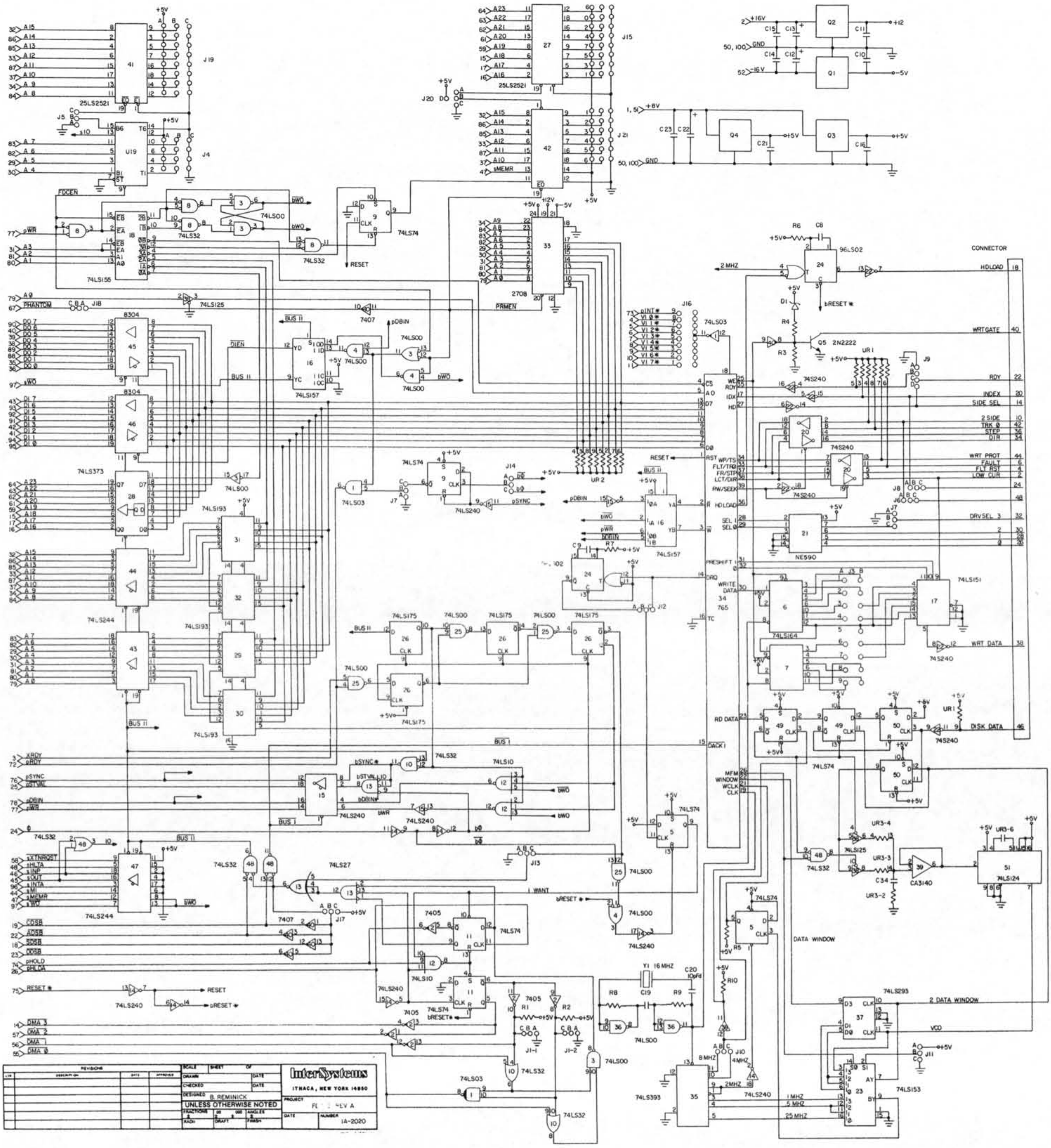
For correct operation with the FDC2, the floppy disk drive should be jumpered to load the heads on active head load. The drive should NOT require active drive select to load the heads.
To select this mode of operation in:

Shugart 800/850, Remex 4000, Qume ; close Jumpers C,X,A
open Jumpers B

Multiple Drives:

For correct operation with the FDC2, the floppy disk drive interface signal lines should only have one pullup resistor per line. Usually this involves making sure that the removable pullup resistor pack be removed from all but one drive in a system. However be careful, some drives do not allow this, especially when mixing drives of different manufacturers together. Often shunt jumpers are provided in the drive to disconnect individual pullup resistors from control lines. When mixing different make drives, it is best to check the manufacturers' documentation.

MANJG3.TXT version 3/05D
edited 4/14/80 3:28 p.m.--jgo edited 4/30/80 12:19 p.m. -- jgo
Second Edition
edited 6/03/80 12:12 a.m. -- jgo
Figure 2.9, 2.15a, 4.1 are changed;
DMA error revisions and applicability
Finney's cxs
parts changes
This symbol -- ™ -- will be a TM when it grows up.



InterSystems
ITHACA, NEW YORK 14850
DESIGNED B. REMINICK
UNLESS OTHERWISE NOTED
PROJECT REV A
NUMBER 1A-2020
CONNECTOR
HDLOAD
WRTGATE
RDY
INDEX
SIDE SEL
2 SIDE
TRK Ø
STEP
DIR
WRT PROT
FAULT
FLT RST
LOW CUR
DRVSEL 3
WRT DATA
DISK DATA
2 DATA WINDOW
VCO
DATA WINDOW
RESET
bRESET *
PHANTOM