FLOPPY DISK CONTROLLER
USERS' MANUAL

NEC MICROCOMPUTERS, INC.

uPD372D

LSI FLOPPY DISK CONTROLLER CHIP

USERS' MANUAL

March 11, 1977

The NEC uPD372D is a high performance N-channel LSI floppy disk controller designed to interface between most minicomputers or microprocessors and most floppy disk drives.

The uPD372D is the most versatile floppy disk controller chip available.

FEATURES – Compatible with IBM3740 format

– Compatible with Shugart Minifloppy format

– Compatible with many other formats

– Generates and checks Cyclic Redundancy Characters

– Initiates operations at address marks or physical index

– Formats clear disks

– Handles up to four floppy disk drives

– Can read or write on one drive while simultaneously track seeking on another

– Track stepping rate and step pulse width are programmable

– Sector size programmable from one byte to one sector per track

– Data transfer rate easily changed

– Standard power supply voltages +12V, +5V, −5V

The uPD372D is compatible with most floppy disk drives including:

Calcomp 140	Pertec FD400
CDC BR803	Potter DD4740
GSI 050	Remex RFS7400
GSI 110	Shugart SA400
Innovex 210	Shugart SA900
Orbis 74	Sycor 145
Persci 75	

TABLE OF CONTENTS

INTRODUCTION

The uPD372 may be interfaced directly to a host processor as shown in Figure 1 or it may be interfaced to a controller processor which in turn is interfaced to the host processor as shown in Figure 2. Most processors interfaced to the uPD372 must, when reading from or writing a record on the disk, be completely dedicated to the task of controlling the disk drive(s). These periods may last several milliseconds. If the nature of the other tasks performed by the host processor allows them to be neglected for this length of time, then the uPD372 may be interfaced directly to the host processor. Otherwise, a controller processor is required.

Whichever approach is chosen, the floppy disk drive control workload is shared between the uPD372 hardware and the processor(s) software.

The uPD372 converts information, which the software transmits to uPD372 internal registers, into commands and serial data which are transmitted to the disk drive(s). The uPD372 converts status signals and serial data from the drives into register information which is read by the software. The uPD372 also controls the timing of most disk operations and performs many other tasks such as Cyclic Redundancy Character generation and checking.

The software tasks are usually divided into two groups as shown in Figures 1 and 2. The Drive Handling Routines control sector reading and writing and track seeking. The Floppy Disk Operating System contains the executive programs which instruct the Drive Handling Routines to read or write a particular sector on a particular track. The Floppy Disk Operating System programs also decide from what area of memory write data should be taken and into what area of memory read data should be stored.

A uPD372 interface between a uPD8080A microprocessor and an IBM3740 compatible drive is shown in the right hand side of Figure 10 in the FLOPPY DISK CONTROLLER EXAMPLE section. This circuit may be interfaced directly to a uPD8080A host processor or to a uPD8080A controller processor similar to the example shown in the left hand side of Figure 10.

Since the disk drive(s) operate under program control, complete uPD8080A assembly listings of the Drive Handling Routines with extensive comments are included in the FLOPPY DISK CONTROLLER EXAMPLE section. The listings describe the sequence of events of each disk operation. Although the sequences are specific to IBM3740 format and to a uPD8080A microprocessor, sequences for other formats are very similar in most respects and the uPD8080A instructions used are largely limited to simple INPUTS, OUTPUTS, CONDITIONAL JUMPS, etc., which are similar to those of other processors.
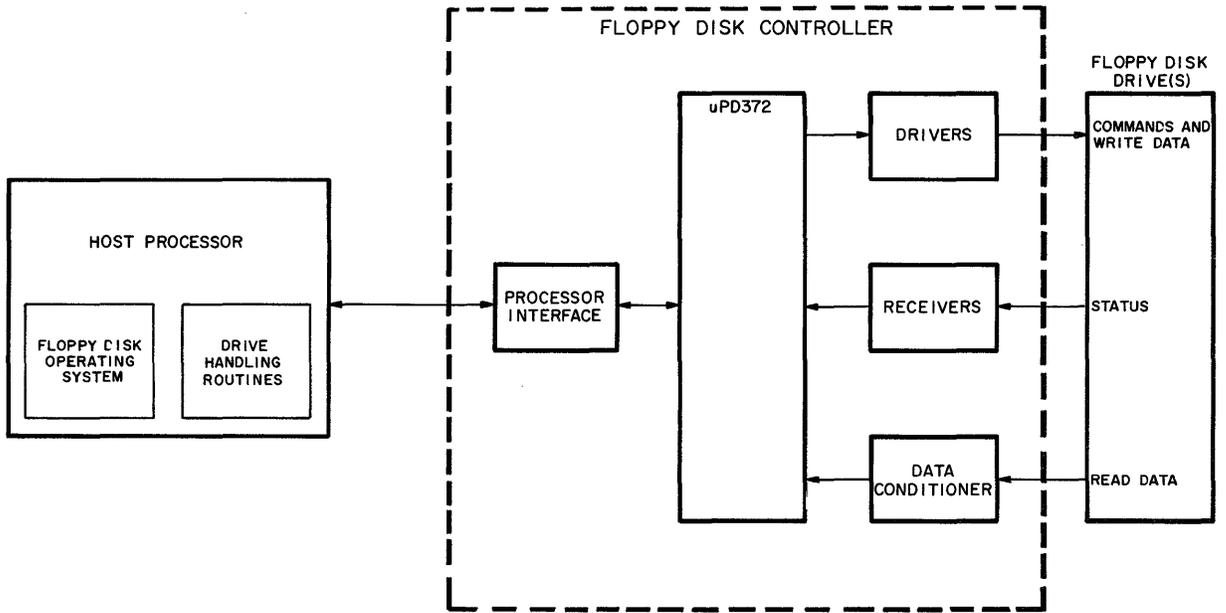
4

FIGURE 1
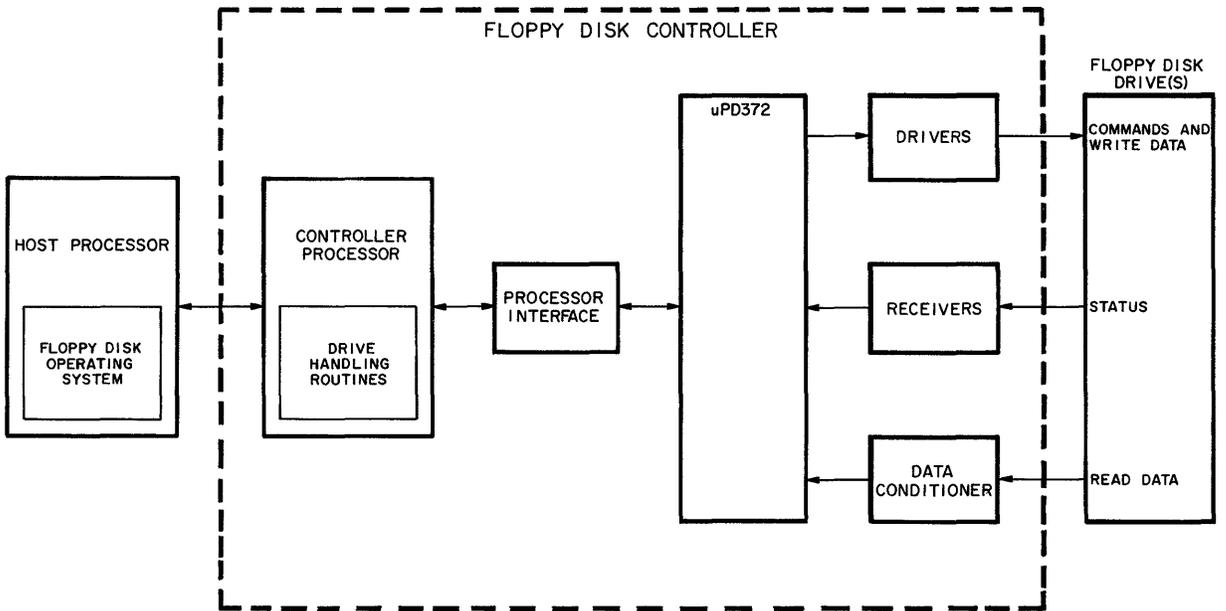uPD372 INTERFACED DIRECTLY TO HOST PROCESSOR (15-20 CHIPS)

FIGURE 2
uPD372 INTERFACED TO HOST PROCESSOR THROUGH CONTROLLER PROCESSOR (34-40 CHIPS)

INTRODUCTION

Figure 3 shows the input/output signals of the uPD372 grouped into 10 categories.

RESET
REGISTER SELECT COMMANDS
INTERRUPT REQUEST
DATA BUS
TIMING
WRITE DATA
DISK DRIVE COMMANDS
READ DATA
DISK DRIVE STATUS
MISCELLANEOUS

The REGISTER SELECT COMMANDS control Data Transfers from the DATA BUS to the 6 uPD372 write registers. The contents of the write registers are translated by the uPD372 into WRITE DATA and DISK DRIVE COMMANDS. The REGISTER SELECT COMMANDS also control Data Transfers from the 3 uPD372 read registers to the data bus. The processor may then read the DISK DRIVE STATUS and READ DATA from the data bus.

The processor generates DISK DRIVE COMMANDS by manipulating bits in the uPD372 write registers and the DISK DRIVE STATUS signals control the state of bits in the read registers. To avoid repetition the signals in these two categories are described in the ADDRESSABLE INTERNAL REGISTERS section only. All other input/output signals are described below.

Processor Interface

Reset

Pin 1 RST (Reset)

A logic one at pin 1 causes a general reset of the uPD372. For a list of signals and registers affected, see RST-bit 7 of Write Register 0 (WR0) in the ADDRESSABLE INTERNAL REGISTERS section.

PROCESSOR INTERFACE

DISK DRIVE INTERFACE

uPD372

RESET → 1 RST

REGISTER WRITE/READ SELECT → 2 W/R
REGISTER DATA STROBE → 3 DS
REGISTER SELECT 2 → 4 RS2
REGISTER SELECT 1 → 5 RS1
REGISTER SELECT 0 → 6 RS0

REGISTER SELECT COMMANDS

INTERRUPT REQUEST ← 17 REQ

WD 18 → WRITE DATA — TO DISK DRIVE

HLD 19 → HEAD LOAD
LCT 22 → LOW CURRENT
WFR 23 → WRITE FAULT RESET
WE 24 → WRITE CURRENT ENABLE
SID 25 → STEP IN OR DIRECTION
SOS 26 → STEP OUT OR STEP
UB1 27 → DISK DRIVE B1 SELECT
UB0 28 → DISK DRIVE B0 SELECT
UA1 29 → DISK DRIVE A1 SELECT
UA0 30 → DISK DRIVE A0 SELECT

DISK DRIVE COMMANDS

DATA BUS 0 ← 31 DB0
DATA BUS 1 ← 32 DB1
DATA BUS 2 ← 33 DB2
DATA BUS 3 ← 34 DB3
DATA BUS 4 ← 35 DB4
DATA BUS 5 ← 36 DB5
DATA BUS 6 ← 37 DB6
DATA BUS 7 ← 38 DB7

DATA BUS

RCK 10 → READ CLOCK
RD 11 → READ DATA

FROM DISK DRIVE

IDX 7 → INDEX
WFT 8 → WRITE FAULT
T00 9 → TRACK ZERO
RYA 12 → DISK DRIVE A READY
RYB 14 → DISK DRIVE B READY

DISK DRIVE STATUS

WRITE CLOCK → 13 WCK
Ø1 → 41 Ø1
Ø2 → 42 Ø2

TIMING

CKS 15 → CLOCK STATUS
AWL 16 → ALWAYS LOW

MISC.

VSS VBB VCC VDD
↑20 ↑21 ↑39 ↑40
GND −5V +5V +12V

FIGURE 3
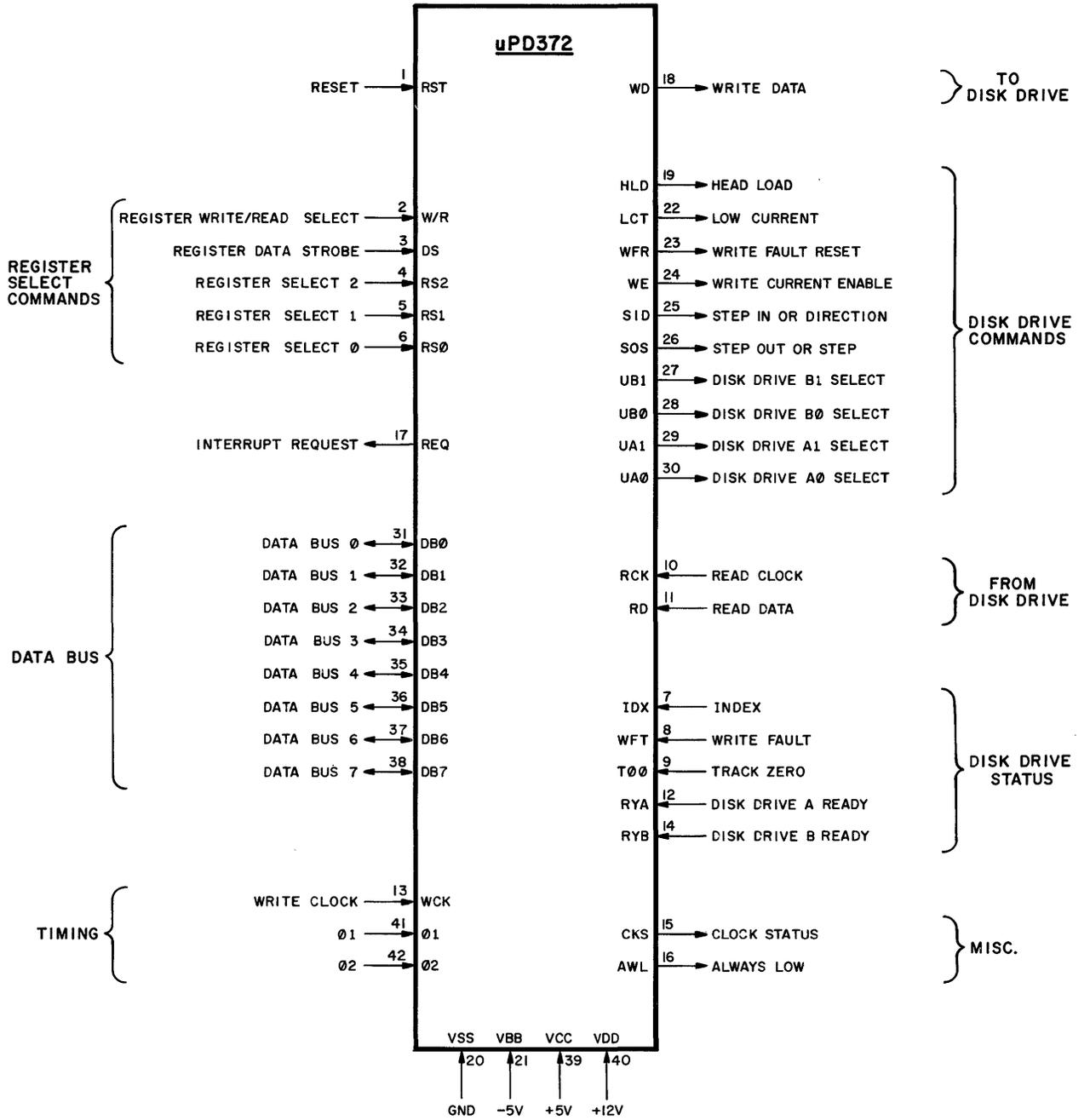**INPUT/OUTPUT SIGNALS**

7

Register Select Commands and Data Bus

    Pin 2   W/R   (Register Write/Read Select)

    Pin 3   DS   (Register Data Strobe)

    Pins 4-6   RS0-RS2   (Register Address)

    Pins 31-38   DB0-DB7   (Data Bus)

       W/R, DS and RS0-RS2 control Data Bus transfers between the uPD372 and the processor as follows:

    Writing into a uPD372 register

        When W/R is a logic one, information the processor
        places on DB0-DB7 is written into the uPD372 Write
        Register selected by RS0-RS2. The information is
        written at the time of the trailing edge of each 02
        which occurs while DS is a logic one.

    Reading from a uPD372 register

        When W/R is a logic zero, information from the uPD372
        Read Register selected by RS0-RS2 is placed on DB0-DB7
        to be read by the processor. The information remains on
        DB0-DB7 as long as DS is a logic one. See uPD372
        SIGNAL TIMING DIAGRAM, Figure 12 for exact timing.

   NOTE: The logic condition which places uPD372 READ
          Register information on the Data Bus is
          DS·$\overline{\text{W/R}}$·$\overline{\text{RS2}}$. If these three signals are
          allowed to change asynchronously with respect
          to each other, care must be taken to insure
          that this condition does not become true in-
          advertantly. The simplest method is to re-
          quire that W/R and RS2 must not change state
          while DS is a logic one.

   NOTE: RS0 and RS1 must not change state during the
          period from 150ns before until 10ns after the
          trailing edge of 01 or else register contents
          and DISK DRIVE COMMANDS may be modified. A
          simple method to accomplish this is to use a
          dual flip-flop to synchronize changes in RS0
          and RS1 with the leading edge of 02(TTL) as
          shown in Figure 10.

Interrupt Request

Pin 17 REQ (Interrupt Request)

Interrupt requests are generated by the uPD372 only while STT (Start-bit 5 of WR3) is true. Two types of interrupt requests occur -- Index Requests and Data Requests.

Index Requests occur once per disk revolution when the physical index hole passes a photodetector. See IRQ-bit 1 of Read Register 0 (RR0) in the ADDRESSABLE INTERNAL REGISTERS section.

Data Requests begin during disk reading after an address mark is read and then occur each time an eight-bit byte, assembled from disk serial data, is available to be read by the processor program. Data requests occur during disk writing or formatting each time an eight-bit byte is required from the processor program. See DRQ-bit 0 of Read Register 0 (RR0) in the ADDRESSABLE INTERNAL REGISTERS section.

Timing Signals

Pin 41 Ø1

Pin 42 Ø2

The uPD372 requires two MOS level clock signals, Ø1 and Ø2. A uPD8224 generates both Ø1 and Ø2 as well as a TTL level Ø2. If the uPD372 is interfaced to a uPD8080A microprocessor and if both devices are in the same proximity, they may share a uPD8224 as shown in Figure 10.

Pin 13 WCK (Write Clock)

WCK determines the bit transfer rate to the selected disk drive while writing. IBM standard drives require a 500KHz WCK. The Shugart Minifloppy requires a 250KHz WCK. Other standards may require different WCK frequencies. In each case the WCK frequency should be twice the bit transfer rate.

DISK DRIVE INTERFACE

Write Data

Pin 18 WD (Write Data)

Serial Frequency Modulated (FM) code to be written on a floppy disk leaves the uPD372 at pin 18. Pin 18 should be connected to the WRITE DATA input of the selected disk drive.

Disk Drive Commands

The following commands to the disk drive(s) are generated under program control by modifying the contents of uPD372 Write Registers. The description of each command may be found in the ADDRESSABLE INTERNAL REGISTERS section under the appropriate write register and register bit.

Pin 19	HLD (Head Load)	See Write Register 0 (WR0) bit 3
Pin 22	LCT (Low Current)	See WR0 bit 2
Pin 23	WFR (Write Fault Reset)	See WR0 bit 1
Pin 24	WE (Write Current Enable)	See WR3 bits 2 and 4
Pin 25	SOS (Step Out or Step)	See WR4 bit 5
Pin 26	SID (Step In or Direction)	See WR4 bit 6
Pin 27	UB1 (Disk Drive B1 Select)	See WR4 bit 1
Pin 28	UB0 (Disk Drive B0 Select)	See WR4 bit 0
Pin 29	UA1 (Disk Drive A1 Select)	See WR1 bit 1
Pin 30	UA0 (Disk Drive A0 Select)	See WR1 bit 0

Read Data

Pin 10 RCK (Read Clock)

Pin 11 RD (Read Data)

RD is a changing logic level updated by each data and clock pulse in the raw data read from the selected disk drive. RCK is a clock the positive transition of which strobes RD into the uPD372. Both signals are derived from the raw data by the DATA CONDITIONER circuit. See the DATA CONDITIONER section for a complete description of RCK and RD.

Disk Drive Status Signals

The following status signals from the disk drive(s) each control the logic level of a bit in uPD372 Read Registers 0 and 1 (RR0 and RR1). By reading the contents of these registers the processor program senses the disk drive status signals. A description of each status signal may be found in the ADDRESSABLE INTERNAL REGISTERS section under the appropriate read register and bit.

Pin 7	IDX (Index)	See Read Register 0 (RR0) bit 1
Pin 8	WFT (Write Fault)	See RR1 bit 2
Pin 9	T00 (Track Zero)	See RR1 bit 6
Pin 12	RYA (Disk Drive A Ready)	See RR1 bit 3
Pin 14	RYB (Disk Drive B Ready)	See RR0 bit 6

Miscellaneous

Pin 15 CKS (Clock Status)

A logic one at CKS indicates that the uPD372 has been commanded (by WCS, bit 6-WR3) to operate with timing signals from the Write Clock (WCK, pin 13). A logic zero at CKS indicates that the uPD372 has been commanded (by RCS, bit 7-WR3) to operate with timing signals from the Read Clock (RCK, pin 10).

Pin 16 AWL (Always Low)

AWL is a logic zero output under all normal operating conditions.

ADDRESSABLE INTERNAL REGISTERS

INTRODUCTION

Data transfers to and from the uPD372 addressable internal registers are controlled by signals W/R, DS and RS0-RS2 at pins 2-6. These signals are discussed in the INPUT/OUTPUT SIGNALS section.

The address of each register and a mnemonic abbreviation for each register bit are shown in Figure 4. The function initiated, controlled or signalled by each register bit is described in this section.

An important internal timing signal, the Bit Ring Pulse (BRP), affects the functions of more than one third of the register bits. The BRP is a pulse that occurs each time 8-bits (one byte) of data have been read from or written on the disk.

While reading:

the first BRP occurs when the first I.D. address mark, Data address mark or Deleted Data address mark is read after STT (bit 5 of WR3) has been set. BRP's continue to occur each time 8 bits (1 byte) have been read until STT is reset.

While writing:

BRP's occur each time 8 bits (1 byte) have been written until STT is reset.

While formating:

the first BRP occurs when the physical index hole passes the floppy disk drive photodetector after IXS and STT (bits 3 and 5 of WR3) have been set. BRP's continue to occur each time 8 bits (1 byte) have been written until STT is reset.

REGISTER ADDRESS				REGISTER NAME	BIT NUMBERS							
W/R	RS2	RS1	RS0		7	6	5	4	3	2	1	0

WRITE REGISTERS

W/R	RS2	RS1	RS0	NAME	7	6	5	4	3	2	1	0
1	0	0	0	WR0	RST	MBL	X	X	HLD	LCT	WFR	X
1	0	0	1	WR1	CBS	X	CB5	CB4	CB3	UAS	UA1	UA0
1	0	1	0	WR2	WD7	WD6	WD5	WD4	WD3	WD2	WD1	WD0
1	0	1	1	WR3	RCS	WCS	STT	WES	IXS	WER	CCG	CCW
1	1	0	0	WR4	STS	SID	SOS	X	X	UBS	UB1	UB0
1	1	1	0	WR6	X	X	X	X	X	TRR	IRR	DRR

READ REGISTERS

W/R	RS2	RS1	RS0	NAME	7	6	5	4	3	2	1	0
0	0	0	0	RR0	ALH	RYB	UB1	UB0	ERR	TRQ	IRQ	DRQ
0	0	0	1	RR1	WRT	T00	DER	COR	RYA	WFT	UA1	UA0
0	0	1	0	RR2	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0

X = NOT USED

FIGURE 4
ADDRESSABLE INTERNAL REGISTERS

13

WRITE REGISTERS

Write Register 0 (WR0)

WR0-Bit 0   Not Used.

WR0-Bit 1   WFR (Write Fault Reset)

A logic one transmitted to bit 1 sets the output of WFR pin 23 to a logic one. The one logic level remains at pin 23 until a logic zero is transmitted to bit 1. The signal at pin 23 may be used for any command purpose, but normally is used to reset a Write Fault condition.

To reset Write Fault, a logic one is transmitted to bit 1 and about 10us later a logic zero is transmitted to bit 1. This forms an approximately 10us wide pulse at pin 23 - the width specified by most disk drive manufacturers.

A Write Fault is signaled by a drive whenever an attempt is made to turn on the write current illegally (viz. when the head is not loaded). The Write Fault condition is sensed by the processor in bit 2 of RR1.

The status of LCT and HLD, bits 2 and 3 of WR0 should be preserved during Write Fault Reset commands. See note after bit 7 of WR0.

WR0-Bit 2   LCT (Low Current)

IBM floppy disk specifications define two values of write current. The higher value is to be used when writing on the outer tracks (00-43) and the lower value is to be used on the inner (higher density) tracks (44-76).

Pin 22, LCT, of the uPD372 should be connected to the floppy disk drive's Low Current input control line. A logic zero in bit 2 sets a flip-flop to the high current state; a logic one sets the flip-flop to the low current state. The output of the flip-flop is connected to pin 22.

During the SEEK routine LCT should be set to a one when the software track counter becomes greater than 43. It should be set to zero when the counter becomes less than or equal to 43.

The state of HLD, bit 3, WR0 should be preserved during Low Current commands. See note after bit 7 of WR0.

WR0-Bit 3   HLD (Head Load)

A logic one at the HLD bit sets the Head Load flip-flop. A logic zero resets the flip-flop. The Q output of the Head Load flip-flop is internally connected to pin 19, HLD. Pin 19 should be connected to the floppy disk drive HEAD LOAD input control line.

The head requires a settling time (approx. 40ms -- see drive specifications) after loading the head and before reading or writing begins.

The state of LCT, bit 2, WR0 should be preserved during Head Load commands. See the note after bit 7 of WR0.

WR0-Bit 4   Not Used.

WR0-Bit 5   Not used.

WR0-Bit 6   MBL (Must Be Low)

This bit must be a logic zero during each command to WR0.

WR0-Bit 7   RST (Reset)

A logic one transmitted to bit 7 resets the uPD372. The effect is exactly the same as that of a pulse on pin 1.

All bits in all write registers; all bits in the read data register, RR2; all disk drive command signals; and Write Data (pin 18) are set to a logic zero.

> NOTE: Commands are made to individual bits in WR0. However, every WR0 bit is affected by any data transfer to WR0. For instance, to load the head of the selected drive, HLD is set to a one, but what should the other bits of the command be? RST, MBL and WFR should be zero. Bits 0, 4 and 5 may be anything but the LCT status of the selected drive should be left unchanged. Consequently, a software image of HLD and LCT must be maintained for each drive. Whenever HLD, LCT or WFR is addressed, the selected drive HLD and LCT status must be recalled and incorporated in the command data transfer. This is accomplished by the WR0 MANAGER subroutine in the software listing for the FLOPPY DISK CONTROLLER EXAMPLE given in this manual.

Write Register 1 (WR1)

WR1-Bit 0   UA0 (Unit A0 Set)

WR1-Bit 1   UA1 (Unit A1 Set)

WR1-Bit 2   UAS (Unit A Strobe)

     Bit 0 (UA0) and bit 1 (UA1) control the logic levels of
output pins 30 (UA0) and 29 (UA1) respectively. These logic
levels may be used for any command function. The logic levels at
pins 30 and 29, along with these at pins 28 and 27 (see WR4 bits
0, 1 and 2 -- UB0, UB1 and UB5) could simply be the select
signals for four disk drives. Or, in a more sophisticated system,
UA0 and UA1 can select one of up to four drives for a read or
write operation and UB0 and UB1 can select another of the drives
for a simultaneous track seek operation.

     If bit 2, UAS is a logic zero when addressing WR1, the logic
levels at pins 30 and 29 remain unchanged regardless of the
content of bits 0 and 1. If bit 2 is a logic one when addressing
WR1, the logic levels at pins 30 and 29 are set to the logic
levels of bits 0 and 1. This allows the write clock bits (bits
3, 4, 5 and 7 of WR1) to be addressed independently from the Unit
A select bits.

WR1-Bit 3   CB3 (Write Clock Bit 3)

WR1-Bit 4   CB4 (Write Clock Bit 4)

WR1-Bit 5   CB5 (Write Clock Bit 5)

WR1-Bit 6   (Not Used)

WR1-Bit 7   CBS (Write Clock Bit Strobe)

     IBM specifications include definitions of four special byte
code patterns termed "address marks". See the INTRODUCTION of
the DATA CONDITIONER section for a discussion of code patterns
and address marks. What distinguishes the address marks from
other bytes sent to or read from the disk is that some of the
"clock" pulses are missing.

     The clock pattern for ID address marks, Data address marks
and Deleted Data address marks is a C7 (hex). The clock pattern
for the Index address mark is D7 (hex). Since all clock pulses
are present for every other byte (gap or data byte) the clock
pattern is FF (hex).

     For all three clock patterns bits 0, 1, 2, 6 and 7 must be
logic ones. Only bits 3, 4 and 5 may be zeros. During writing
the uPD372 allows the state of clock bits 3, 4 and 5 to be

16

controlled by CB3, CB4 and CB5. The other clock bits are always logic ones.

In order to change the state of CB3, CB4 or CB5, CBS must be a logic one. If CBS is a zero, CB3, CB4 and CB5 are unaffected by data transfers into WR1.

To set an FF (hex) clock pattern for data and gap bytes, B8 (hex) should be written into WR1. To set a C7 (hex) clock pattern for ID, Data or Deleted Data address marks, 80 (hex) should be written into WR1. To set a D7 (hex) clock pattern for Index address marks, 90 (hex) should be written into WR1.

Write Register 2 (WR2)

WR2-Bits 0-7 WD0-WD7 (Write Data Register)

When the uPD372 is writing on the disk, the data from WR2 and the clock pattern from WR1 are transferred to a 16-bit shift register once every 16 Write Clock (WCK) cycles. The data and clock bits are frequency modulation encoded and serially transmitted from the shift register through pin 18, WD (Write Data) to the selected floppy disk drive read/write head.

If the processor does not transfer new data to WR2, the write data pattern in WR2 remains constant and is repeatedly written on the disk.

Write Register 3 (WR3)

WR3-Bit 0 CCW (Cyclic Check Words)

CCW must be set to a one while the floppy disk drive read/write head is either reading or writing the last data byte of an ID or data record. It must be reset to a zero while the head is reading or writing the second CRC byte. See program listings for READ ID, READ DATA and WRITE DATA for exact sequence.

In the Read Mode, the Bit Ring Pulse (BRP) which follows the setting of CCW begins a bit-by-bit serial comparison between the CRC bytes read from the disk and the CRC bytes generated and stored in the uPD372 CRC register. The comparison ends at the time of the BRP which follows the resetting of CCW. The same BRP sets bit 5 of RR1 (DER-Data Error) to a one if an error was detected. DER is set to a zero if no error was detected. The DER information remains valid for one byte time (while the head is reading the first gap byte following the record).

In the Write mode, CCW is used to write the two CRC bytes. The BRP which follows the setting of CCW causes write data to be

taken from the CRC register rather than the write data register, WR2. The writing of the CRC bytes ends at the time of the BRP following the resetting of CCW. See program listing for WRITE DATA.

WR3-Bit 1 CCG (Write Cyclic Check Generator Start)

CCG is used only in the write mode to start the calculation of the CRC. (CRC calculation begins automatically at the address mark in the read mode.) CCG must be set to a one while the head is writing the last gap byte before writing an ID or Data record. CRC calculation begins at the time of any BRP which occurs while CCG is a one. Consequently, CCG must be reset to a zero while the head is writing the first byte (address mark) of the record or the next BRP would start the CRC calculation again. See the program listing for WRITE DATA for the exact sequence.

WR3-Bit 2 WER (Write Enable Reset)

See Bit 4 of WR3 (Write Enable Set--WES)

WR3-Bit 3 IXS (Index Start)

IXS is used in conjunction with STT to start the formatting routine at the physical index. See STT, bit 5 of WR3.

WR3-Bit 4 WES (Write Enable Set)

A logic one at WES sets and a logic one at WER resets a flip-flop. The Q output of this flip-flop is synchronized with the BRP to set the Write Enable flip-flop. The Q output of the Write Enable flip-flop is connected to pin 24, WE. The signal on this pin should be used to enable the write current of the selected floppy disk drive.

WR3-Bit 5 STT (Start)

STT starts and ends every read, write and format operation.

Reading:

An A0 (hex) is transmitted to WR3 (STT and RCS=1). The next ID address mark, Data address mark or Deleted Data address mark read by the disk drive causes an interrupt request and a BRP. Interrupts continue to occur as each byte is read and at each physical index until STT is reset.

Writing:

Each write operation is preceded by a READ ID routine
(which STT starts). STT remains a logic one during
READ ID, during the gap between the ID record and the
data record and while writing a new data record. STT
is not reset until the data record has been written.

The clock source is switched to the Write Clock and
write current is enabled 11 bytes (in IBM format) after
the ID record. Writing begins at the next BRP. Six
new gap bytes are written, the data record is written
and one new byte in the following gap is written before
STT is reset. After STT is reset, the next (and last)
BRP resets the write current.

BRP´s and interrupt requests occur at each byte from
the start of READ ID until STT is reset at the end of
WRITE DATA. See listing of WRITE DATA.

Formatting:

A 78 (hex) is transmitted to WR3 (STT, WCS, WES and IXS
= 1). The next physical index signal (IDS pin 7)
enables the write current and starts a series of BRP´s
and interrupt requests. These continue for one entire
disk revolution while the track is being formatted with
data bytes, gap bytes and address marks provided by the
processor. STT is reset at the next index request flag
(IRQ bit 1 of RR0) ending the BRP´s and interrupt
requests and turning off the write current. See the
listing of the FORMAT routine.

Resetting STT automatically resets Write Enable, sets WCS
(Write Clock Set) and prevents further data requests (DRQ) and
index requests (IRQ). See program listings of READ ID, READ DATA
and WRITE DATA for examples of the use of STT.

WR3-Bit 6   WCS (Write Clock Set)

WR3-Bit 7   RCS (Read Clock Set)

WCS and RCS do not enable write current nor do they determine
whether the uPD372 is in the write mode or the read mode; they
simply select one of two sources of clock signals for the
internal timing of the uPD372. The two clock sources are WCK
(Write Clock-pin 13) and RCK (Read Clock-pin 10). Both signals
have a frequency of 500KHz (2usec period) for IBM format. WCK
should be derived from a crystal controlled oscillator. RCK must
be derived from and synchronized with data and clock pulses read
from a disk. This is accomplished by the DATA CONDITIONER
circuit. See DATA CONDITIONER section.

The Read Clock is used only when STT is set and data is being read from the disk. The Write Clock is the normal clock source. The timing source is switched from the Read Clock to the Write Clock by any of the following.

1. By the BRP following the setting of WCS.

2. By resetting STT

3. By RST

The logic level of CKS (Clock Status) pin 15 is set high by any of the above.

The source of timing signals is switched from WCK to RCK by setting RCS. The logic level of CKS is set low by RCS.

Write Register 4 (WR4)

WR4-Bit 0 UB0 (Unit B0 Select)

WR4-Bit 1 UB1 (Unit B1 Select)

WR4-Bit 2 UBS (Unit B Strobe)

Bit 0 (UB0) and bit 1 (UB1) control the logic levels of output pins 28 (UB0) and 27 (UB1) respectively. These logic levels may be used for any command function. The logic levels at pins 28 and 27, along with those at pins 30 and 29 (see WR1 bits 0, 1 and 2--UA0, UA1 and UAS) could simply be the select signals for four disk drives. Or, in a more sophisticated system, UA0 and UA1 can select one of up to four drives for a read or write operation and UB0 and UB1 can select another of the drives for a simultaneous track seek operation.

If bit 2, UBS is a logic zero when addressing WR4, the logic levels at pins 28 and 27 remain unchanged regardless of the content of bits 0 and 1. If bit 2 is a logic one when addressing WR4, the logic levels at pins 28 and 27 are set to the logic levels of bits 0 and 1. This allows the step bits (bits 5, 6 and 7 of WR4) to be addressed independently from the unit B select bits.

WR4-Bit 3 Not used

WR4-Bit 4 Not used

WR4-Bit 5   SOS (Step Out or Step)

WR4-Bit 6   SID (Step In or Direction)

WR4-Bit 7   STS (Step Strobe)

    Bits 5 (SOS) and 6 (SID) control the logic levels of output
pins 26 (SOS) and 25 (SID) respectively. These logic levels may
be used for any command function but are normally used to form
stepping pulses to move the read/write head of the selected
drive.

    The step pulse width, repetition rate and the direction
control scheme vary from one drive to another. The uPD372 can
adapt to any drive with software changes only. For instance,
many drives require a DIRECTION logic level to determine which
direction (in or out) the read/write head is to move and a STEP
pulse to trigger the actual movement. For these drives uPD372
pin 25 is connected to the drive DIRECTION input and pin 26 to
the STEP input. To move the read/write head, DIRECTION is set
with bit 6 and STEP is set to a one with bit 5. One pulse width
later (usually about 10usec) STEP is set to a zero with bit 5.
One repetition period later (usually about 10msec) STEP is again
set to a one with bit 5. One pulse width later STEP is set to a
zero again. This process continues until the read/write head
arrives at the correct track.

    The timing of the repetition rate is facilitated by the
1.024msec timer provided by the uPD372. See RR0-Bit 2, TRQ
(Timer Request).

    Other drives require STEP IN pulses on one input line and
STEP OUT pulses on another. For these drives uPD372 pin 25 is
connected to the drive STEP IN input and pin 26 is connected to
the drive STEP OUT input. STEP IN or STEP OUT pulses are formed
under program control in a manner similar to that described
above.

    If bit 7, STS (Step Strobe) is a logic zero when addressing
WR4, the logic levels at pins 26 and 25 remain unchanged
regardless of the content of bits 5 and 6. If bit 7 is a logic
one when addressing WR4, the logic levels of pins 26 and 25 are
set to the logic levels of bits 5 and 6. This allows the unit
select bits (bits 0, 1 and 2 of WR4) to be addressed
independently from the step bits.

Write Register 5 (WR5)   Not used.

Write Register 6 (WR6)

WR6-Bit Ø   DRR (Data Request Reset)

      A logic one transmitted to DRR resets the Data Request, DRQ.
   See DRQ bit Ø of RRØ.

WR6-Bit 1   IRR (Index Request Reset)

      A logic one transmitted to IRR resets the Index Request
   (IRQ). See IRQ bit 1 of RRØ.

WR6-Bit 2   TRR (Timer Request Reset)

      A logic one transmitted to TRR resets the Timer Request
   (TRQ). See TRQ bit 2 of RRØ.

WR6-Bit 3-Bit 7   Not Used.

READ REGISTERS

Read Register Ø (RRØ)

RRØ-Bit Ø   DRQ (Data Request)

      When DRQ is true, the processor controlling the uPD372
   should read a data byte from RR2 during the read mode or transmit
   a data byte to WR2 during the write mode.

      DRQ causes an Interrupt Request (REQ) at pin 17.

      DRQ´s are generated as follows:

      1.   During the read mode the first ID address mark, Data
           address mark or Deleted Data address mark (but not
           Index address mark) which is read following the setting
           of STT (bit 5-WR3) sets DRQ. From this point on, DRQ
           is set again by every BRP (i.e., after every byte--data
           byte, gap byte or address mark--is read) until STT is
           reset. DRQ must be reset, by DRR (Data Request
           Reset--bit Ø, WR6) each time that it is set. If DRQ is
           still true at the time of the following BRP, a Command
           Overrun Error results (signaled by COR--bit 4, RR1).
           DRQ is automatically reset when STT is reset.

      2.   During the write mode Data Requests occur at each BRP
           (i.e., after each byte--data byte, gap byte or address
           mark--is written). As in (1) above, DRQ must be reset
           each time that it is set or a Command Overrun Error
           results.

3. During formatting a series of DRQ's begins when the
    physical index hole is detected after setting STT and
    IXS (bits 3 and 5 of WR3). The DRQ's continue to
    occur at every BRP until STT is reset. (STT and IXS
    should be reset at the end of one complete
    revolution--when IRQ, Index Request, bit 2, RR6 becomes
    true.)

RR0-Bit 1 IRQ (Index Request)

     IRQ is set true by the leading edge of the physical index
pulse. The physical index pulse is generated when the index hole
of the floppy disk passes a photo detector in the disk drive.
STT must be set to enable IRQ.

     IRQ causes an Interrupt Request (REQ--pin 17).

     IRQ is reset by transmitting a logic one to IRR (Index
Request Reset--bit 1, WR6), by resetting STT and by RST.

RR0-Bit 2 TRQ (Timer Request)

     TRQ is the Q output of a flip-flop which is set by every
512th Write Clock (WCK, pin 13) pulse. The Write Clock period
for IBM compatible controllers is 2us causing TRQ to be set every
1.024msec. TRQ is set every 2.048msec when using Shugart
Minifloppy format.

     TRQ does not cause an Interrupt Request (REQ, pin 17).

     TRQ is reset by transmitting a logic one to TRR (Timer
Request Reset--bit 2, WR6) and by RST.

RR0-Bit 3 ERR (Error)

     ERR indicates a condition that must be corrected before
issuing a command to the disk drive. ERR is the logical OR of
three status signals:

$$ERR = WFT + \overline{RYA} + COR$$

where: WFT is Write Fault--bit 2, RR1

        RYA is Disk Drive A Ready--bit 3, RR1

        COR is Command Overrun--bit 4, RR1

$\overline{RYB}$ is not involved in the calculation of ERR.

23

RRØ-Bit 4 UBØ (Drive BØ Selected)

RRØ-Bit 5 UB1 (Drive B1 Selected)

        UBØ and UB1 are two of the four status bits (the other being
    bits Ø and 1 of RR1) that indicate which disk drive has been
    selected. See descriptions of bits Ø, 1 and 2 of WR4 and pins 27
    and 28.

RRØ-Bit 6 RYB (Drive B Ready)

        This status bit indicates the logic level of RYB at pin 14.
    Pin 14 is usually connected to the READY output of the drive
    selected by UBØ and UB1 (bits Ø and 1 of WR4).

RRØ-Bit 7 ALH (Always High)

        This bit is a logic one as long as power is supplied to the
    uPD372.

Read Register 1 (RR1)

RR1-Bit Ø UAØ (Drive AØ Selected)

RR1-Bit 1 UA1 (Drive A1 Selected)

        UAØ and UA1 are two of the four status bits (the other being
    bits 4 and 5 of RRØ) that indicate which disk drive has been
    selected. See descriptions of bits Ø, 1 and 2 of WR1 and pins 29
    and 3Ø.

RR1-Bit 2 WFT (Write Fault)

        This status bit indicates the logic level of WFT, pin 8.
    Pin 8 is usually connected to the WRITE FAULT output of the
    selected drive. A write fault condition occurs when a floppy
    disk drive detects an illegal command during a write operation.
    All commands to that drive are ignored as long as the write fault
    condition exists. The write fault is reset by WFR, bit 1, WRØ.

RR1-Bit 3 RYA (Drive A Ready)

        This status bit indicates the logic level of RYA at input
    pin 12. Pin 12 is usually connected to the READY output of the
    drive selected by UAØ and UA1 (bits Ø and 1 of WR1).

RR1-Bit 4   COR (Command Overrun)

        COR indicates that the processor did not respond in time to
a Data Request (DRQ) during either a read or a write operation.
See DRQ, bit 0 of RR0.

RR1-Bit 5   DER (Data Error)

        DER indicates that a CRC error occurred during a read
operation. DER is explained in detail in the description of CCW
(Cyclic Check Words) bit 0 of WR3.

RR1-Bit 6   T00 (Track Zero)

        This status bit indicates the logic level of T00 at input
pin 9. Pin 9 is usually connected to the TRACK 00 output of the
selected disk drive. The disk drive places a high logic level on
TRACK 00 when and only when the read/write head is at track zero.

RR1-Bit 7   WRT (Write Mode)

        WRT indicates which clock signals the uPD372 is using for
internal timing--Write Clock (WCK, pin 13) or Read Clock (RCK,
pin 10). The selection is made by WCS (Write Clock Select) or
RCS (Read Clock Select) bits 6 and 7 of WR3.

        The logic level of WRT is identical to the logic level of
CKS, output pin 15.

Read Register 2 (RR2)

RR2-Bits 0-7   RD0-RD7 (Read Data Register)

        Data, serially read from the selected disk drive, is
assembled into 8-bit parallel bytes in an internal shift register
and is then transferred to RR2 at each BRP. See READ ID and READ
RECORD routines for examples of the use of the Read Data
Register.

DATA CONDITIONER

INTRODUCTION

Frequency modulation encoded data as transmitted to or received from a floppy disk drive consists of a series of timing or "clock" pulses interleaved with a series of data pulses (see Figure 5). The period between successive clock pulses is termed a "data cell". The presence of a data pulse within a data cell represents a logic one data bit. The absence of a data pulse within a data cell represents a logic zero data bit.

The clock pulses also contain information. Just as with the data pulses the presence of a clock pulse within a data cell represents a logic one clock bit. And, the absence of a clock pulse within a data cell represents a logic zero clock bit. Each byte is made up of eight data bits and eight clock bits for a total of sixteen.

Since the data read from a floppy disk is in serial format, the controller must have some means of distinguishing between clock bits and data bits and must also be able to determine the beginning of each byte. To provide this synchronization, soft-sectored floppy disks are written with a special byte, an address mark, at the beginning of every record.

All bytes, except address marks, contain the full complement of eight clock pulses for a clock pattern of FF (hex). In IBM format the address marks written at the beginning of records (ID Address Mark, Data Address Mark and Deleted Data Address Mark) each have a C7 (hex) or 11000111 (binary) clock pattern.

The data pattern for an ID address mark is FE (hex) or 11111110 (binary), for a Data address mark is FB (hex) or 11111011 (binary) and for a Deleted Data address mark is F8 (hex) or 11111000 (binary).

The generalized sixteen bit (interleaved clock and data) pattern for these address marks takes the form:

11110101011A1B1C (binary)

where at least one of the numbers, A, B or C is a zero.

This pattern cannot be generated by any sixteen bit sequence read from a floppy disk except those of an address mark. All other sixteen bit sequences must have a logic one in alternate locations to represent an FF (hex) clock pattern.
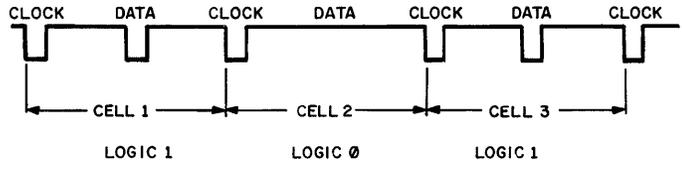
FIGURE 5
FLOPPY DISK DRIVE SERIAL DATA FORMAT

$Tcy = \emptyset 1$ CYCLE TIME
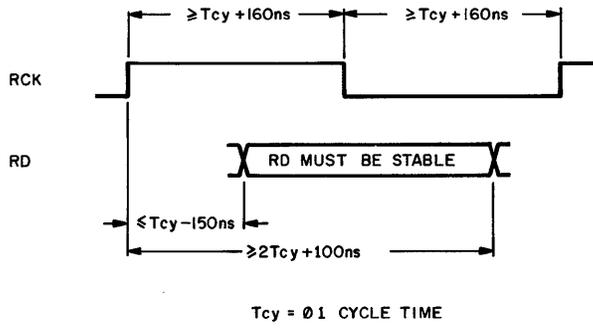
FIGURE 6
READ CLOCK (RCK) AND READ DATA (RD) REQUIRED BY uPD372
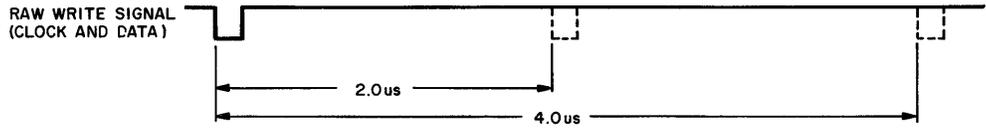
FIGURE 7
IBM FORMAT WRITE DATA TRANSMITTED TO DRIVE
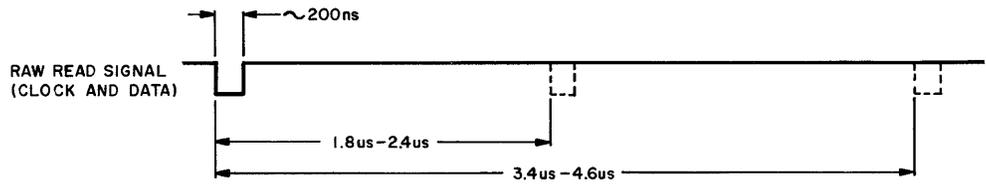
FIGURE 8
SAME DATA READ BACK (TRACK 76)

27

Each bit (clock and data) received by the uPD372 from a floppy disk drive is clocked into a sixteen bit shift register. Once STT (bit 5 of register WR3) is set to a one, the uPD372 begins looking for the address mark pattern in the shift register. When the address mark pattern is found, a BRP (Bit Ring Pulse) is formed, the eight data bits in the shift register are transferred into the Read Register (RR2) and the Data Request flag is raised. From that point until STT is reset, every sixteenth bit clocked into the shift register causes a BRP and a Data Request and transfers the data bits to the Read Register.

Signals Required by the uPD372

In order to clock a bit into the uPD372 shift register, the desired logic level, zero or one, must be maintained at the READ DATA input (RD, pin 11) and a positive going transition must be made at the READ CLOCK input (RCK, pin 10).

The uPD372 samples RCK with the trailing edge of Ø2. Since RCK is asynchronous with Ø2, there is an uncertainty of one Ø2 clock cycle, Tcy, in the time when the uPD372 senses the positive going edge of RCK. Furthermore, RCK is not sampled instantaneously -- a setup time and hold time are required. Consequently, after RCK goes high, it must remain high for at least Tcy + 160ns to guarantee that the positive transition is sensed. For the same reason, after RCK returns to the low state, it must remain low for Tcy + 160ns before the next positive transition.

Once a positive transition of RCK is sensed by Ø2, the following Ø2 clocks the logic level at RD into the shift register. So, the earliest time that RD can be clocked is nearly Tcy after the positive transition of RCK and the latest time is about 2 Tcy after the positive transition of RCK. RD must be stable during this period and for a short time before and after.

The timing requirements of RCK and RD are summarized in Figure 6.

Most floppy disk controllers require that the data pulses read from a drive be separated from the clock pulses by an external "DATA SEPARATOR" circuit. The uPD372, on the other hand, reads the combined clock and data information and separates the data internally. So, it does not require an external DATA SEPARATOR.

The uPD372 does, however, require some modification of the raw data read from the disk. The modification is accomplished by a circuit termed a "DATA CONDITIONER".

28

Data Conditioner Algorithm

The function of the DATA CONDITIONER is to translate the pulses of the raw data, read from a disk, into the RD and RCK signals required by the uPD372. Except for an amendment, which for the sake of clarity is introduced later, the DATA CONDITIONER performs the translation by using the following algorithm:

1. Each time a raw data pulse (clock or data) is received from a floppy disk drive, the DATA CONDITIONER sets the uPD372 RD input to a logic one and sends a positive pulse to the RCK input. The RD and RCK signals must meet the requirements of Figure 6. In addition, the receipt of a raw data pulse starts an interval timer with a period of 1.4T--where T is the average time between adjacent clock and data pulses. In other words, T is the approximate time before the next raw data pulse will arrive, if it is present. GO TO 2. or 3.

2. If the next raw data pulse arrives before 1.4T, the DATA CONDITIONER continues to maintain a logic one at RD, generates another RCK pulse and restarts the interval timer. GO TO 2. or 3.

3. If no raw data pulse has arrived by 1.4T, the pulse is declared absent. The DATA CONDITIONER sets the RD input to a logic zero and generates another RCK pulse. The interval timer is not reset until the next raw data pulse is received. GO TO 1.

An amendment must be made to the above algorithm in order to guarantee that the requirements of Figure 6 are always satisfied. Once RCK goes high, it must remain high for a least Tcy + 160ns before going low. Once low, it must remain low for at least Tcy + 160ns before going high. The minimum RCK cycle time is, therefore, 2Tcy + 320ns. For the normal Tcy of 500ns the minimum RCK cycle time is 1320ns. This presents no problem in the long run because the average time between adjacent raw data pulses (in IBM format) is 2us, but it does present a problem between any two closely spaced pulses.

Figure 7 shows the regularly spaced raw write data sent to a floppy disk drive and Figure 8 shows the timing variations in the same data read back from the drive. A decision that the second pulse is missing in Figure 8 cannot be made until 2.8us after the first pulse (T = 2us in IBM format). If a RCK cycle is begun at 2.8us, it cannot be completed for another 1320ns or not until 4.12us after the first pulse; but, the third pulse may occur as early as 3.4us. Obviously, in order to meet the requirements of Figure 6, the DATA CONDITIONER must be capable of storing the information about one raw data pulse while completing the RCK cycle of another.

To meet this requirement the first sentence of the algorithm
must be amended as follows:

1. Each time a raw data pulse (clock or data) is received
   from a floppy disk drive, the DATA CONDITIONER waits
   until the previous RCK cycle is completed and then sets
   the uPD372 RD input to a logic one and sends a positive
   pulse to the RCK input.

DATA CONDITIONER EXAMPLE

A DATA CONDITIONER which uses the above algorithm and with
one-shot times set for IBM format is shown in Figure 9 along with
a timing diagram. Tcy, the Ø2 cycle time, is assumed to be
500ns.

Raw data consisting of clock and data pulses is shown in the
top line of the timing diagram. The first pulse occurs at time
zero. The second pulse is missing indicating a logic zero. The
third pulse is early by 600ns. The fourth pulse occurs at the
expected time. This pattern is the worst case because of the
short time interval between the decision that the second pulse is
missing and the early third pulse.

One-shot U31A generates the 1.4T time interval. Flip-flop
U29A stores the information that a RCK cycle should begin when
possible. One-shot U31B prevents the start of a RCK cycle until
the previous cycle is completed. Flip-flop U29B double buffers
the raw data. One-shot U30B forms RCK.

Other Formats and Other Ø2 Cycle Times

The same DATA CONDITIONER will operate with formats other
than IBM and with Ø2 cycle times other than 500ns.

When using other formats the only parameter which affects
the DATA CONDITIONER is the read data transfer rate and the only
one-shot affected is the 1.4T time interval one-shot, U31A. The
period of U31A should be 1.4T for all T. For example: T=2us in
IBM Format so the period of U31A should be 2.8us and T=4us for
the Shugart Minifloppy so the period of U31A should be 5.6us.

When using other Ø2 cycle times, the two one-shots which
define RCK are affected. If the Ø2 cycle time is Tcy, then the
period of U30B should be Tcy + 250 ns and the period of U31B
should be 2 Tcy + 500 ns in order to keep RCK consistent with
Figure 6.
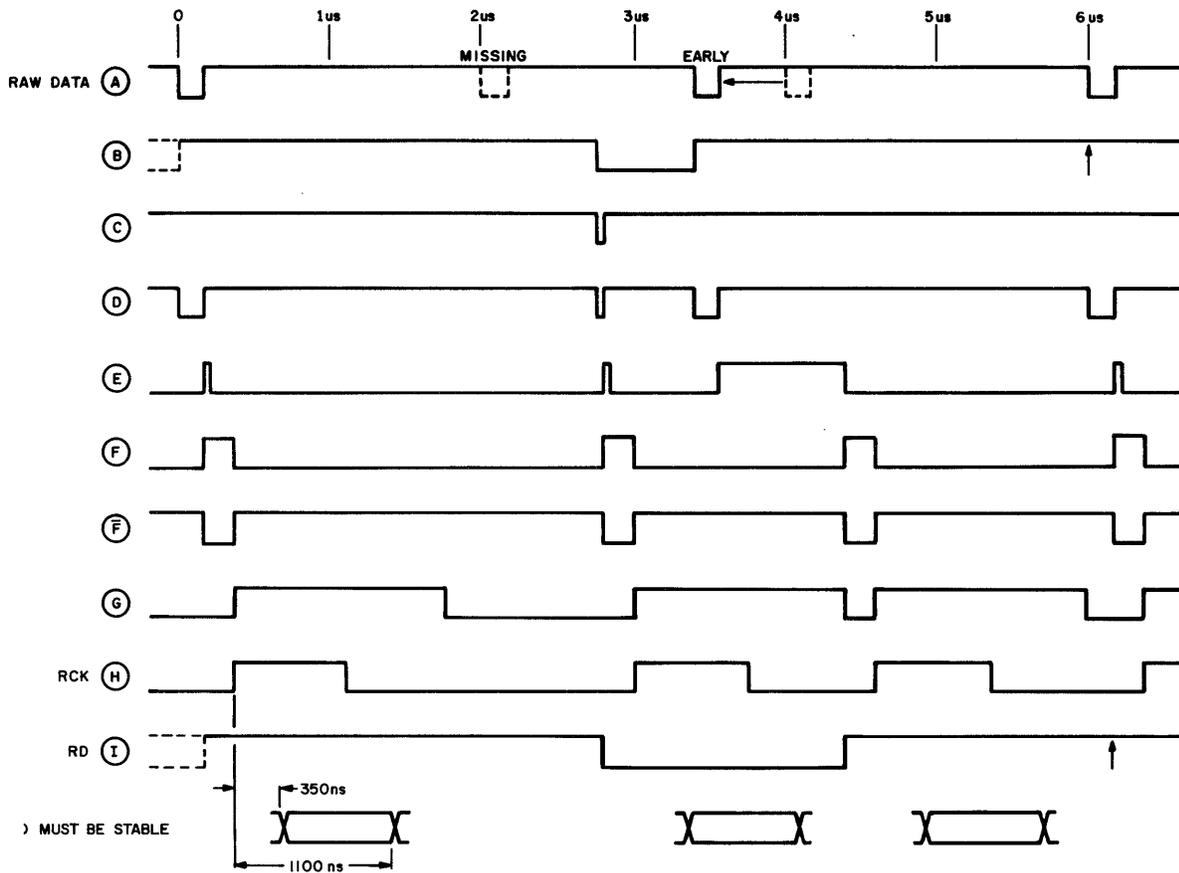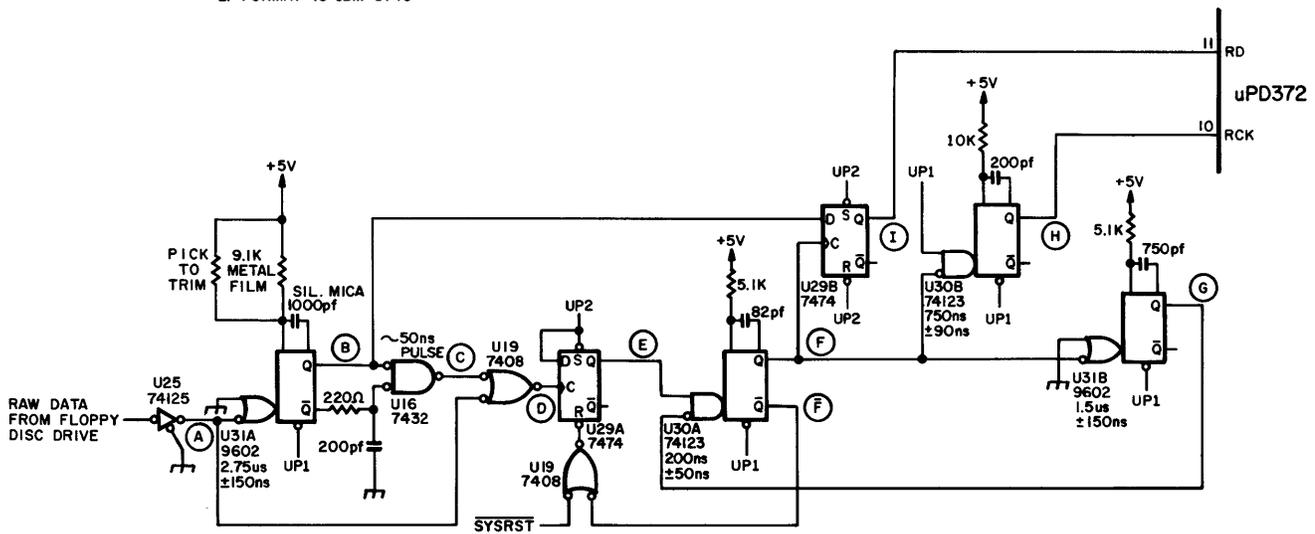
FIGURE 9
DATA CONDITIONER EXAMPLE AND TIMING DIAGRAM

FLOPPY DISK CONTROLLER EXAMPLE

HARDWARE

Figure 10 is divided into two sections. The section on the right shows a floppy disk controller which may be interfaced either directly to a host processor or indirectly through a controller processor. The section on the left shows a controller microprocessor system. Both the host and the controller processors are uPD8080A's in this example and the single floppy disk drive is IBM3740 compatible.

The controller and host processors share a common memory. 1K bytes of ROM is accessed by the controller processor between 0000-03FF of its address space. The same ROM is accessed by the host processor between E000-E3FF of its address space. 256 bytes of RAM is accessed by the controller processor between 0400-04FF of its address space. The same RAM is accessed by the host processor between E400-E4FF of its address space. The ROM stores the Disk Handling Routines and the RAM provides temporary storage for data, commands, status, etc.

The controller processor system configuration shown in Figure 10 is a testbed for software and hardware changes in the floppy disk interface. The intent is to give the host processor (which is assumed to have an editor and a console) the ability to monitor the performance of the controller processor and the rest of the floppy disk interface. Consequently, the direction of Direct Memory Access is from the host into the controller rather than vice versa and no provision is made to allow the controller to interrupt the host.

An interrupt feature would require the addition of only two chips. The resulting controller would be satisfactory for many applications although the host would have to transfer data between the common RAM and the desired storage locations. A circuit which allows Direct Memory Access from the controller into the host would require several logic changes resulting in a net gain of a few chips.

Circuit Description

The circuitry to the right of the uPD372 in Figure 10 contains the line drivers and receivers for the floppy disk drive. One shot U26B "stretches" the 60-100ns wide Write Data (WD) pulses into the width specified by the drive manufacturer.

The logic below the uPD372 is the Data Conditioner which is described in the DATA CONDITIONER section.
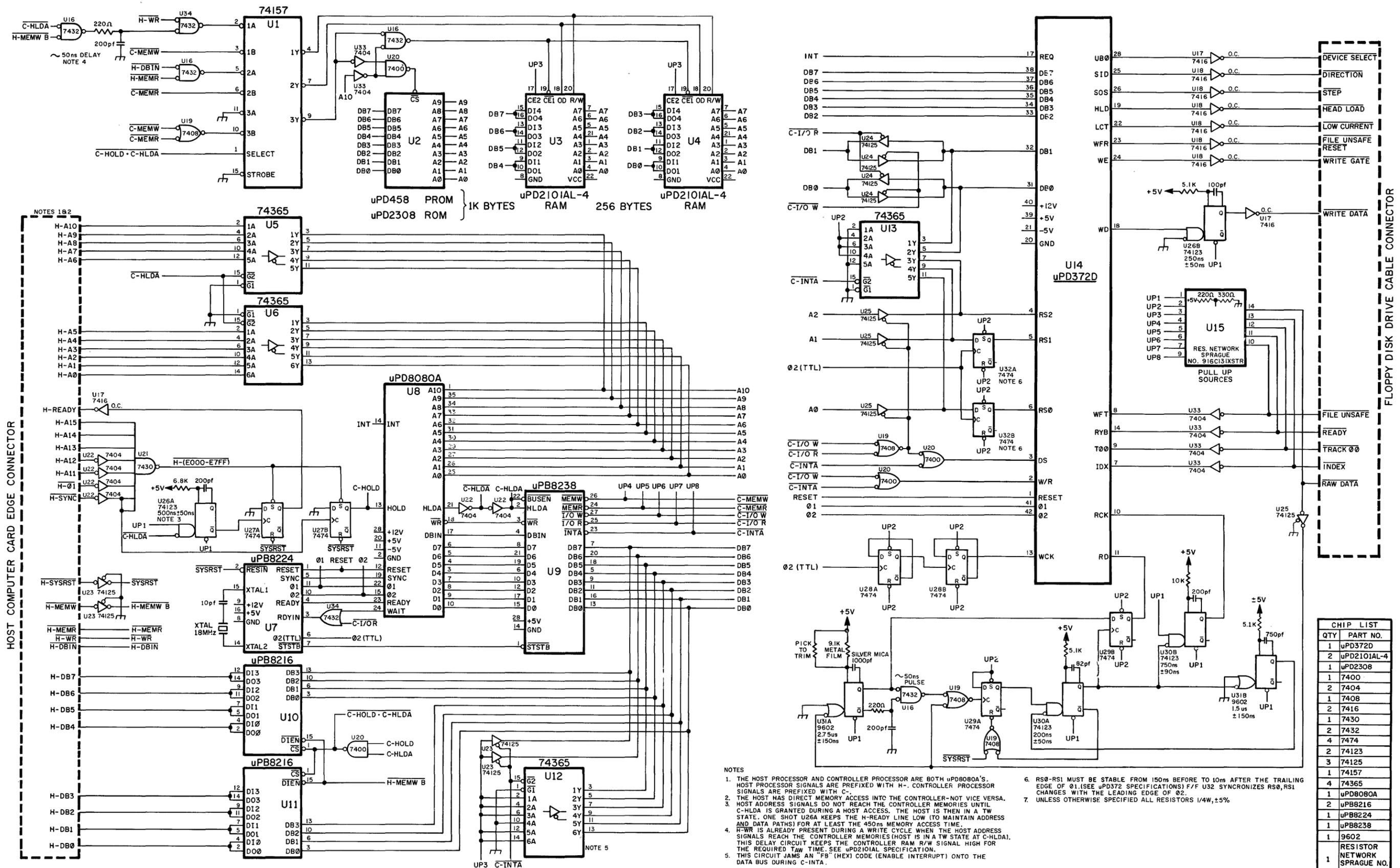
FIGURE 10 - uPD372D FLOPPY DISC INTERFACE

3-11-77

CHIP LIST		
QTY	PART NO.	
1	uPD372D	
2	uPD2101AL-4	
1	uPD2308	
1	7400	
2	7404	
1	7408	
2	7416	
1	7430	
2	7432	
4	7474	
2	74123	
3	74125	
1	74157	
4	74365	
1	uPD8080A	
2	uPB8216	
1	uPB8224	
1	uPB8238	
1	9602	
1	RESISTOR NETWORK SPRAGUE NO. 914C131XSTR	
34	TOTAL	

NOTES
1. THE HOST PROCESSOR AND CONTROLLER PROCESSOR ARE BOTH uPD8080A'S. HOST PROCESSOR SIGNALS ARE PREFIXED WITH H-. CONTROLLER PROCESSOR SIGNALS ARE PREFIXED WITH C-.
2. THE HOST HAS DIRECT MEMORY ACCESS INTO THE CONTROLLER-NOT VICE VERSA.
3. HOST ADDRESS SIGNALS DO NOT REACH THE CONTROLLER MEMORIES UNTIL C-HLDA IS GRANTED DURING A HOST ACCESS. THE HOST IS THEN IN A TW STATE. ONE SHOT U26A KEEPS THE H-READY LINE LOW (TO MAINTAIN ADDRESS AND DATA PATHS) FOR AT LEAST THE 450ns MEMORY ACCESS TIME.
4. H-WR IS ALREADY PRESENT DURING A WRITE CYCLE WHEN THE HOST ADDRESS SIGNALS REACH THE CONTROLLER MEMORIES (HOST IS IN A TW STATE AT C-HLDA). THIS DELAY CIRCUIT KEEPS THE CONTROLLER RAM R/W SIGNAL HIGH FOR THE REQUIRED Taw TIME. SEE uPD2101AL SPECIFICATION.
5. THIS CIRCUIT JAMS AN "FB" (HEX) CODE (ENABLE INTERRUPT) ONTO THE DATA BUS DURING C-INTA.
6. RS0-RS1 MUST BE STABLE FROM 150ns BEFORE TO 10ns AFTER THE TRAILING EDGE OF Ø1. (SEE uPD372 SPECIFICATIONS) F/F U32 SYNCRONIZES RS0,RS1 CHANGES WITH THE LEADING EDGE OF Ø2.
7. UNLESS OTHERWISE SPECIFIED ALL RESISTORS 1/4W,±5%

33

Flip-flops U28A and U28B generate a 500KHz Write Clock for IBM formats. The Shugart Minifloppy format requires an additional flip-flop to generate a 250KHz clock. Other formats may require other clock frequencies. In each case the period of the clock is equal to the period between adjacent clock and data pulses in the Frequency Modulated data stream.

U13, U24 and 3 gates of U25 force a 01 (hex) into WR6 during interrupt acknowledge. This causes a hardware reset of the Data Request (but not of the Index Request). See bits 0 and 1 of WR6.

The remaining chips in the right hand section of Figure 10, give a uPD8080A processor data transfer control over the uPD372 addressable internal registers.

The Direct Memory Access control logic is shown at the left of the uPD8080A in Figure 10. Flip-flops U27A and U27B are both set at the beginning of each host machine cycle in which the host addresses the common memory. Flip-flop U27B makes a HOLD request of the controller uPD8080A. Flip-flop U27A places the host uPD8080A in the WAIT state by lowering the host's READY input. When the controller uPD8080A grants the HOLD request, signified by C-HLDA, the host uPD8080A is given control over:

1) the Chip Enable, Output Disable and Read/Write inputs of the common memory by U1;

2) the common memory address lines by U5 and U6;

3) the common memory data bus by U10 and U11.

After one-shot U26A causes a delay equal to the memory access time, flip-flop U27A releases the host uPD8080A from the WAIT state. When the host uPD8080A has completed the machine cycle in which it addressed common memory, it issues a SYNC pulse which causes U27B to release the controller uPD8080A from the HOLD state. When C-HLDA goes low the controller processor is free to continue execution of its own programs.

There is sufficient delay before C-HLDA goes low to prevent this design from allowing two successive host machine cycles to be addressed to the common memory. The main effect is to prevent the host processor from executing programs in common memory, which is not a requirement of this design.

OR-gate U34, between the uPD8080A and the uPD8224, causes one wait state to occur in each machine cycle during which data is read from the uPD372; but does not cause a wait state in any other machine cycle. The uPD372 places read register information on the data bus 90ns (max.) after 01 – which is too late to remain within uPD8080A specifications without a wait state. A wait state is not required at any other time. In fact a wait state in every machine cycle is not allowed since it would make a uPD8080A system too slow to handle disk data transfers.

34

SOFTWARE

The software for the controller processor may be divided
into catagories as follows:

Executive Routines

> INITIALIZE
> RETURN FROM COMMAND
> IDLE LOOP
> EXECUTE COMMAND

Main Routines

> READ ID
> READ DATA
> WRITE DATA
> FORMAT
> SEEK

Subroutines

> STEP IN
> STEP OUT
> HEAD LOAD
> HEAD UNLOAD
> WRØ MANAGER
> UNIT SELECT
> DELAY

The Executive Routines initialize the floppy disk drive
system after startup, interpret commands from the host processor
and transfer control to the appropriate Main Routine. The Main
Routines perform the actual disk drive operations. At the
completion of a Main Routine, control is transferred back to the
Executive Routines which inform the host processor of the result
and then wait for the next processor command.

All of the software, including FORMAT, for a single disk
drive controller may be stored in one 1K x 8 ROM. Nearly 1/4 of
this area is available for program expansion to allow control of
up to four disk drives and for other purposes. This area is made
available by storing most of FORMAT on a disk rather than in ROM.
When formatting is required, the major portion of FORMAT is
recalled from a disk and stored in the controller RAM with only a
minor portion permanently stored in ROM.

An introduction is given below to READ ID, READ DATA and
WRITE DATA. The comments in the uPD8080A assembler listing which
follows this section should be sufficient to understand the rest
of the controller software.

Read Sector M on Track N

Before the read operation begins, the host processor writes the desired sector address in SECTR (location 403 in the listing) the desired track address in TRACK, the desired disk drive in UNIT and the sector size in SCTSZ. If no sector size is transmitted from the host, the controller program assumes a size of 128 (decimal) bytes.

The host processor initiates the read operation by writing an 01 in CMND (location 400). The controller program interprets the 01 as a read command and if the sector address, track address, drive address and sector size are within acceptable limits, control of the controller processor is transferred to READ DATA (See READ DATA in listing).

READ DATA starts by calling READ ID to find the correct sector and track.

READ ID

READ ID starts by calling the SEEK routine which positions the read/write head at the requested track. Next the read/write head is loaded (if it is not already loaded because of a previous operation). The read/write head then begins reading somewhere on the requested track. See IBM TRACK FORMAT Figure II. A limit is placed on the number of index holes (4) that may be counted without successfully reading the specified ID record before giving up.

The H, L registers are initialized to point to TRACK, the location containing the desired track number and the first location in the group of parameters.

The D, E registers are initialized to point to WTRK (Wrong Track) the first location in a group of error flags. uPD8080A register B is cleared and a 00 is transmitted to uPD372 write register WR3. The effect is to insure that one particular bit, bit 5, STT (Start) is at a logic zero. All the other bits of WR3 are known to be at a logic zero at this time.

An A0 is then transmitted to WR3. The effect of this command is to set STT, bit 5, and RCS (Read Clock Set) bit 7. RCS instructs the uPD372 to use read clock pulses from the disk for its internal timing control. STT instructs the uPD372 to cause an interrupt request when the next address mark passes the read/write head.

Interrupts are then enabled and the processor is halted while waiting for the interrupt.
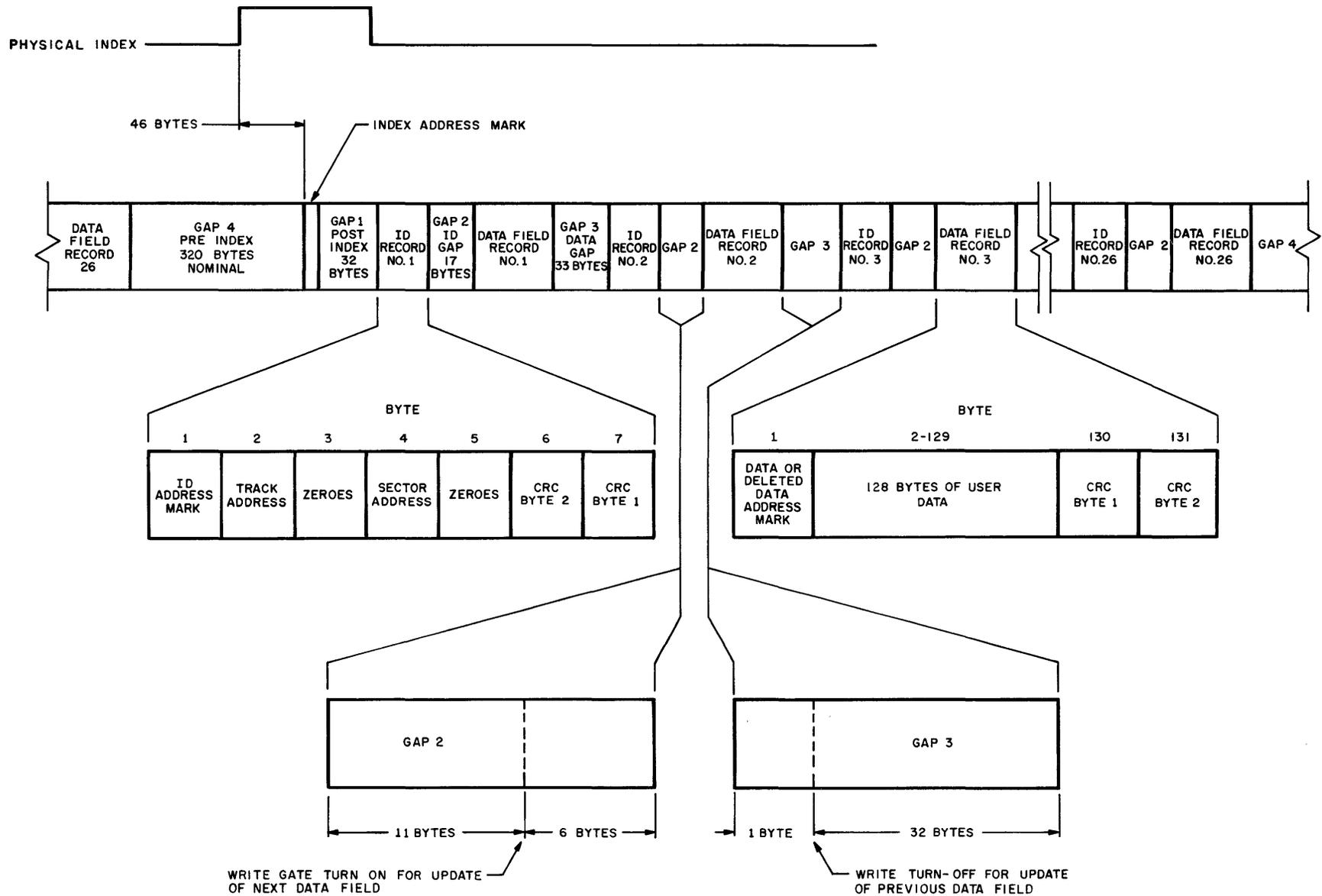
FIGURE 11
IBM TRACK FORMAT

37

Halt -- Interrupt Technique

The transfer rate of IBM standard floppy disks is one byte every 32us which is too fast for the usual technique of "jamming" a RST (Restart) instruction on the data bus during INTA (Interrupt Acknowledge) followed by a JMP to a service routine which ends with a RET (Return) instruction.

Instead, as shown in the READ ID routine, the processor is halted before an expected interrupt to keep the Program Counter from advancing.

When an address mark is read, an interrupt request is generated and CRC calculation automatically begins.

During INTA an EI (Enable Interrupts) instruction is jammed onto the data bus (by U12 and U23 shown in the bottom center of Figure 10) and is executed by the processor. The Program Counter is not incremented by an instruction which occurs during INTA so the next instruction executed is the one following the HLT.

Another event which occurs during each INTA is the clearing of the data request flag by U13, U24 and U25 (near the center of Figure 10) which write a 01 into WR6 (DRR-Data Request Reset).

After the address mark interrupt and the EI instruction, the READ ID routine inputs the data byte read from the disk. The data is compared with FE (hex) to determine whether or not the address mark which caused the interrupt was an ID address mark. If it was not an ID address mark, a JMP is made to the RIM subroutine which makes sure the revolution limit has not been exceeded, and then returns the Program Counter to RIA to wait for the next address mark. If it was an ID address mark, the processor is halted to wait for the next interrupt.

The next interrupt occurs when the uPD372 has read the first data byte of the ID record -- the track address. The processor is then halted again.

Using the halt-interrupt technique the READ ID routine reads the first zero byte, the sector address, the second zero byte and the two CRC bytes, storing appropriate error messages if required. If the sector address is incorrect, the Program Counter is returned to RIA and READ ID waits for the next address mark.

The sector number of the ID record just read provides enough information for a READ ID program to compute what the delay will be before the desired ID record will come around to the read/write head. This time could be spent by the controller· processor to perform other tasks such as track seeking on another drive.

Eventually, an ID record is read with the correct sector address. If no read errors occur, READ ID returns control to READ DATA having found Sector M and Track N.

READ ID exited having read three of the 17 gap bytes between the ID record and the data record (See Figure 11). Between gap bytes 11 and 12, there may be noise on the disk resulting from the turning on of write current each time the data record was updated in the past. To avoid encountering the noise, READ DATA switches the uPD372 internal timing to the Write Clock. Using the Write Clock which has the same frequency as the Read Clock (although the two are not in sync) READ DATA times out past the noisy area.

After the 13th byte (approximately) STT is reset. Several "housekeeping" functions are then performed in preparation for reading the data record.

Near the center of the 6 "clean" bytes adjacent to the data record, STT is set and the controller processor is halted to wait for the data address mark.

When the address mark is read, an interrupt occurs and the CRC calculation begins. If the address mark was a Deleted Data mark or an illegal address mark, appropriate error flags are set. If the address mark was a Data mark, then the data record is read and stored in the data buffer in the common RAM (locations 480-4FF in the controller processor address space).

If no CRC error is found, READ DATA returns control to the Executive Routines which signal the host processor that the requested data is in the common RAM.

Write Sector M on Track N

Before the write operation begins, the host processor fills the data buffer in common RAM and writes the desired sector, track and drive addresses and sector size in the parameter area.

The host processor initiates the write operation by writing 02 in CMND. The controller processor Executive Routines then transfer control to WRITE DATA (See WRITE DATA in listing).

WRITE DATA begins in exactly the same way as READ DATA by calling READ ID to find the correct sector and track. READ ID returns control to WRITE DATA after reading three gap bytes.

WRITE DATA continues reading the gap bytes until gap byte 11 is read. At the end of gap byte 11, an attempt is made to turn on the write current. If the disk drive indicates a Write Fault condition, the write current is not turned on and WRITE DATA is aborted.

If no Write Fault is indicated, 6 gap bytes, all the data

bytes, the two CRC bytes and one gap byte are written. The Write current is then turned off, STT is reset and control is returned to the Executive Routines which signal the host processor that the data buffer has been written in the requested sector of the requested track.

```
        ; NEC FLOPPY DISK DRIVE CONTROLLER PROGRAM 03-09-77 1100 (GCY)
        ; CALLING SEQUENCE
        ;   SET UNIT(1-NU), TRACK(0-NTRKS), SECTR(1-NSCTR)
        ;   OPTIONAL:  SET SCTSZ(4-NBSCT)
        ;         SET CMND(1-NCMDS) LAST
        ;         WAIT FOR CMND=0
        ;
        ; EQUATES FOR USE WITH UPD372
        ;
0000    W0      EQU 0     ;WRITE REGISTER ZERO
0080    W0RST   EQU 80H   ;RESET
0040    W0MBL   EQU 40H   ;MUST BE LOW
0008    W0HLD   EQU 08H   ;HEAD LOAD
0004    W0LCT   EQU 04H   ;LOW CURRENT
0002    W0WFR   EQU 02H   ;WRITE FAULT RESET
        ;
0001    W1      EQU 1     ;WRITE REGISTER ONE
0080    W1CBS   EQU 80H   ;CLOCK BIT STROBE
0038    W1CBN   EQU 38H   ;CLOCK BITS FOR NORMAL DATA
0010    W1CBI   EQU 10H   ;CLOCK BITS FOR INDEX ADDRESS MARK
0000    W1CBD   EQU 00H   ;CLOCK BITS FOR ID, DATA,
                          ;OR DELETED DATA ADDRESS MARK
0004    W1UAS   EQU 04H   ;UNIT A STROBE
0003    W1UAA   EQU 03H   ;UNIT A ADDRESS MASK
        ;
0002    W2      EQU 2     ;WRITE DATA REGISTER
        ;
0003    W3      EQU 3     ;WRITE REGISTER THREE
0080    W3RCS   EQU 80H   ;READ CLOCK SET
0040    W3WCS   EQU 40H   ;WRITE CLOCK SET
0020    W3STT   EQU 20H   ;START READ/WRITE OPERATION
0010    W3WES   EQU 10H   ;WRITE ENABLE SET
0008    W3IXS   EQU 08H   ;INDEX START
0004    W3WER   EQU 04H   ;WRITE ENABLE RESET
0002    W3CCG   EQU 02H   ;CYCLIC CHECK GENERATE
0001    W3CCW   EQU 01H   ;CYCLIC CHECK WORDS
        ;
0004    W4      EQU 4     ;WRITE REGISTER FOUR
0080    W4STS   EQU 80H   ;STEP STROBE
0040    W4SID   EQU 40H   ;STEP IN OR DIRECTION
0020    W4SOS   EQU 20H   ;STEP OUT OR STEP
0004    W4UBS   EQU 04H   ;UNIT B STROBE
0003    W4UBA   EQU 03H   ;UNIT B ADDRESS MASK
        ;
0006    W6      EQU 6     ;WRITE REGISTER SIX
0004    W6TRR   EQU 04H   ;TIMER REQUEST RESET
0002    W6IRR   EQU 02H   ;INDEX REQUEST RESET
0001    W6DRR   EQU 01H   ;DATA REQUEST RESET
        ;
0000    R0      EQU 0     ;READ REGISTER ZERO
0080    R0ALH   EQU 80H   ;ALWAYS HIGH
0040    R0RYB   EQU 40H   ;READY B
0030    R0UBA   EQU 30H   ;UNIT B ADDRESS MASK
```

```
0008            R0ERR    EQU 08H  ;ERROR
0004            R0TRQ    EQU 04H  ;TIMER REQUEST
0002            R0IRQ    EQU 02H  ;INDEX REQUEST
0001            R0DRQ    EQU 01H  ;DATA REQUEST
                ;
0001            R1       EQU 1    ;READ REGISTER ONE
0080            R1WRT    EQU 80H  ;WRITE MODE
0040            R1T00    EQU 40H  ;TRACK 00
0020            R1DER    EQU 20H  ;DATA ERROR (CRC)
0010            R1COR    EQU 10H  ;COMMAND OVERRUN
0008            R1RYA    EQU 08H  ;READY A
0004            R1WFT    EQU 04H  ;WRITE FAULT
0003            R1UAA    EQU 03H  ;UNIT A ADDRESS MASK
                ;
0002            R2       EQU 2    ;READ DATA REGISTER
                ;
0000            MU       EQU 0                  ;MULTIPLE UNITS (0=NO, 0FFFFH=YES)
0001            NU       EQU 1                  ;NUMBER OF FDD UNITS
004D            NTRKS    EQU 77                 ;NUMBER OF TRACKS
001A            NSCTR    EQU 26                 ;NUMBER OF SECTORS
0080            NBSCT    EQU 128                ;NUMBER OF BYTES IN A SECTOR
0003            NTRYS    EQU 3                  ;NUMBER OF READ RETRYS
002B            LHCTK    EQU 43                 ;LAST HIGH CURRENT TRACK
0004            RVLIM    EQU 4                  ;REVOLUTION LIMIT IN IDLE LOOP
0470            STACK    EQU 0470H
                ;
                ;
0000                     ORG 0
0000 F3         RST0:    DI                     ;RESET
0001 3E80                MVI A,W0RST
0003 D300                OUT W0                 ;RESET 372
0005 317004              LXI SP,STACK
                ;INITIALIZE DATA AREA TO ZERO
0008 210004     RS020:   LXI H,CMND             ;HL=ADR(DATA AREA)
000B 0617                MVI B,NB               ;B=NO. OF BYTES
000D AF                  XRA A
000E 77         RS030:   MOV M,A                ;M=0
000F 23                  INX H
0010 05                  DCR B                  ;DONE?
0011 C20E00              JNZ RS030              ;NO
                ; INITIALIZE SECTOR SIZE
0014 3E80                MVI A,NBSCT
0016 320404              STA SCTSZ
                ; INITIALIZE ALL UNITS
                         IF MU
                         MVI A,1                ;START WITH UNIT #1
                RS010:   STA UNIT               ;SET UNIT #
                         ENDIF
0019 CD1F00              CALL INIT
                         IF MU
                         LDA UNIT
                         CPI NU                 ;LAST UNIT?
                         JZ RT010               ;YES
                         INR A                  ;NO
```

42

```
                        JMP RS010          ;GO DO NEXT UNIT
                        ENDIF
001C C34600             JMP RT010
                ;
                ;
                ;INITIALIZE DISK UNIT SUBROUTINE
001F            INIT    EQU $              ;INITIALIZE WR0
                        IF MU
                        LXI D,WR0
                        CALL INDXA
                        ENDIF
                        IF NOT MU
001F 211304             LXI H,WR0
                        ENDIF
0022 AF                 XRA A
0023 77                 MOV M,A            ;SET WR0=0
0024 CDB202             CALL UNLD          ;UNLOAD HEAD
                ; MOVE HEAD TO TRACK ZERO
0027 0E4C               MVI C,NTRKS-1      ;C=LOOP LIMIT
0029 DB01      IN010:   IN R1              ;READ STATUS
002B E640               ANI R1T00          ;TRACK 0?
002D C23700             JNZ IN020          ;YES, DONE
0030 CD6C02             CALL STO           ;NO, STEP OUT
0033 0D                 DCR C
0034 C22900             JNZ IN010          ;CHECK AGAIN
0037           IN020    EQU $
                        IF MU
                        LXI D,TKPTR
                        CALL INDXA
                        ENDIF
                        IF NOT MU
0037 211404             LXI H,TKPTR
                        ENDIF
003A AF                 XRA A
003B 77                 MOV M,A            ;TKPTR=0
003C C9                 RET
                ;
                ; RETURN FROM COMMAND
003D F3        RETRN:   DI
003E CA4600             JZ RT010           ;WAS THERE AN ERROR?
0041 3E01               MVI A,1            ;YES, SET THE
0043 320504             STA MERF           ;MASTER ERROR FLAG
0046 3E20      RT010:   MVI A,W3STT        ;NO
0048 D303               OUT W3             ;ACTIVATE INDEX REQUESTS
                        IF NOT MU
004A 211504             LXI H,REVS
                        ENDIF
                        IF MU
                        LXI D,REVS
                        CALL INDXA
                        ENDIF
004D AF                 XRA A
004E 77                 MOV M,A            ;SET IDLE REVS TO ZERO
004F 3E03               MVI A,NTRYS
```

```
0051 321204           STA RRTRY             ;RESET NO. OF READ RETRYS
0054 AF               XRA A
0055 320004           STA CMND              ;RESET COMMAND TO ZERO
              ;
              ; IDLE LOOP - CHECK FOR A COMMAND
0058 3A0004   IDL10:  LDA CMND
005B B7               ORA A
005C C27D00           JNZ EXEC              ;EXECUTE COMMAND
              ; NO COMMAND, UPDATE IDLE REVS
                      IF MU
                      MVI C,NU              ;YES, UPDATE IDLE REVS
              IDL20:  MOV A,C
                      STA UNIT
                      CALL UNSLC
                      LXI D,REVS
                      CALL INDXA
                      ENDIF
                      IF NOT MU
005F CDD002           CALL UNSLC
0062 211504           LXI H,REVS
                      ENDIF
0065 DB00             IN R0                 ;READ STATUS
0067 E602             ANI R0IRQ             ;INDEX REQUEST?
0069 CA7A00           JZ IDL30              ;NO
006C D306             OUT W6                ;YES, IRQ RESET
006E 34               INR M
006F 7E               MOV A,M               ;A=REVS
0070 FE04             CPI RVLIM             ;ARE REVS<RVLIM?
0072 FA7A00           JM IDL30              ;YES, CONTINUE
0075 AF               XRA A                 ;NO
0076 77               MOV M,A               ;REVS=0
0077 CDB202           CALL UNLD             ;UNLOAD HEAD
007A          IDL30   EQU $
                      IF MU
                      DCR C                 ;DONE?
                      JNZ IDL20             ;NO, CHECK NEXT UNIT
                      ENDIF
007A C35800           JMP IDL10             ;YES, CHECK FOR COMMAND
              ; EXECUTE COMMAND
              ;   A=COMMAND(1-N)
007D 4F       EXEC:   MOV C,A               ;SAVE COMMAND IN C
007E 110504           LXI D,MERF            ;ZERO FLAGS
0081 060D             MVI B,NF
0083 AF               XRA A
0084 12       EX005:  STAX D
0085 13               INX D
0086 05               DCR B
0087 C28400           JNZ EX005
008A 3E06             MVI A,NCMDS
008C B9               CMP C                 ;IS CMND OK?
008D F29600           JP EX010              ;YES
0090 320604           STA CMDER             ;NO, SET FLAG
0093 C3C600           JMP ERROR
              ; CHECK ALL PARAMETERS
```

```
0096 110104    EX010:  LXI  D,UNIT        ;DE=ADR(PARAMETERS)
0099 21CF00            LXI  H,LMTBL       ;HL=ADR(LIMIT TABLE)
009C 0604             MVI  B,NP          ;B=NO. OF PARAMETERS
009E 1A       EX020:  LDAX D             ;A=PARAMETER
009F BE               CMP  M             ;LOWER LIMIT OK?
00A0 FAC100           JM   EX040         ;NO, ERROR
00A3 23               INX  H             ;YES
00A4 BE               CMP  M             ;UPPER LIMIT OK?
00A5 F2C100           JP   EX040         ;NO, ERROR
00A8 23               INX  H             ;YES
00A9 13               INX  D
00AA 05               DCR  B             ;DONE?
00AB C29E00           JNZ  EX020         ;NO
                   ; COMMAND AND PARAMETERS OK
00AE 79               MOV  A,C           ;YES, A=COMMAND
00AF 21D700           LXI  H,CTBL
00B2 3D               DCR  A             ;A=(0-(N-1))
00B3 07               RLC                ;A=2*A
00B4 5F               MOV  E,A
00B5 1600             MVI  D,0
00B7 19               DAD  D             ;HL=ADR(ADR)
00B8 5E               MOV  E,M
00B9 23               INX  H
00BA 56               MOV  D,M           ;DE=ADR
00BB EB               XCHG               ;HL=ADR
00BC 113D00           LXI  D,RETRN
00BF D5               PUSH D             ;(SP)=RETURN ADDRESS
00C0 E9               PCHL               ;JUMP TO ROUTINE
                   ;
00C1 3E01     EX040:  MVI  A,1
00C3 320704           STA  PRMER         ;SET PARAMETER FLAG
                   ;
00C6 320504    ERROR:  STA  MERF          ;SET MASTER ERROR FLAG
00C9 317004           LXI  SP,STACK      ;RESET SP
00CC C34600           JMP  RT010
                   ;
                   ;LIMIT TABLE (UPPER AND LOWER FOR PARAMETERS)
00CF 0102     LMTBL:  DB 1,NU+1          ;UNIT
00D1 004D             DB 0,NTRKS         ;TRACK
00D3 011B             DB 1,NSCTR+1       ;SECTR
00D5 0481             DB 4,NBSCT+1       ;SCTSZ
                   ;
                   ; COMMAND TABLE
00D7 6901     CTBL:   DW READ            ;1
00D9 E601             DW WRITE           ;2
00DB E300             DW SEEK            ;3
00DD 1F00             DW INIT            ;4
00DF F202             DW FRMAT           ;5
00E1 0000             DW RST0            ;6
0006          NCMDS   EQU ($-CTBL)/2
                   ;
                   ;
                   ; SEEK TRACK ROUTINE
                   ;   REGISTERS: A,F,B,DE,HL
```

```
                    ;     SUBROUTINES: INDXA,STI,STO
                    ;
00E3              SEEK      EQU $
                            IF NOT MU
00E3 211404                 LXI H,TKPTR
                            ENDIF
                            IF MU
                            LXI D,TKPTR
                            CALL INDXA         ;HL=ADR(TRACK POINTER)
                            ENDIF
00E6 3A0204                 LDA TRACK          ;A=TRACK DESIRED
00E9 BE                     CMP M
00EA C8                     RZ                 ;TRACK=TKPTR
00EB FAF400                 JM SK010           ;TKPTR>TRACK
00EE CD6202                 CALL STI           ;TKPTR<TRACK
00F1 C3E300                 JMP SEEK
00F4 CD6C02      SK010:     CALL STO           ;TKPTR>TRACK
00F7 C3E300                 JMP SEEK
                    ;
                    ; READ ID RECORD ROUTINE
                    ;
                    ;
                    ;REGISTERS: A,F,B,C,DE,HL
                    ;
00FA CDE300      RID:       CALL SEEK          ;POSITION HEAD
00FD CD9A02                 CALL HDLD          ;LOAD HEAD
                    ;
0100 0E04                   MVI      C,4       ;STORE LIMIT OF REVOLUTIONS OF
                                               ;DISK WITHOUT FINDING CORRECT ID
                                               ;RECORD.  USE 4 TO GUARANTEE
                                               ;THREE COMPLETE REVOLUTIONS.
                    ;
0102 210204      RIA:       LXI H,TRACK        ;INITIALIZE TRACK/SECTR POINTER
0105 110A04                 LXI D,WTRK         ;INITIALIZE FLAG POINTER
0108 AF                     XRA      A
0109 47                     MOV B,A            ;SET B=0
010A D303                   OUT W3             ;RESET STT (FOR RETRY)
                                               ;
010C 3EA0                   MVI A,W3RCS+W3STT ;RCS=1, STT=1
010E D303                   OUT      W3        ;GO TO READ CLOCK.  SET STT TO AUTO-
                                               ;MATICALLY START READ OPERATION WHEN
                                               ;ADDRESS MARK IS READ.
                                               ;
0110 FB                     EI                 ;ENABLE INTERRUPT AND WAIT FOR
0111 76                     HLT                ;ADDRESS MARK TO BE READ.
                    ;
                    ;INTERRUPT(ADDRESS MARK)
                    ;
                    ;         (EI)
0112 DB02                   IN       R2        ;READ DATA
0114 EEFE                   XRI      0FEH      ;IS IT AN I.D. ADDRESS MARK?
0116 C25201                 JNZ      RIM       ;NO: JUMP TO RIM
0119 76                     HLT                ;YES: WAIT FOR NEXT INTERRUPT.
                    ;
```

```
                    ;INTERRUPT(TRACK ADDRESS)
                    ;
                    ;            (EI)
011A DB02           IN       R2        ;READ TRACK ADDRESS BYTE.
011C AE             XRA M              ;COMPARE WITH DESIRED TRACK
011D 12             STAX D             ;WTRK =0 FOR OK, =NON-ZERO FOR ERROR
011E 13             INX D              ;DE POINTS TO ZERO1
011F 76             HLT                ;WAIT FOR NEXT INTERRUPT
                    ;
                    ;INTERRUPT(FIRST ZERO BYTE)
                    ;
                    ;            (EI)
0120 DB02           IN       R2        ;READ ZERO BYTE
0122 12             STAX D             ;ZERO1 =0 FOR OK, =NON-ZERO FOR ERROR
0123 13             INX D              ;DE POINTS TO ZERO2
0124 23             INX H              ;HL POINTS TO SECTR
                                       ;
0125 76             HLT                ;WAIT FOR NEXT INTERRUPT
                    ;
                    ;INTERRUPT(SECTOR ADDRESS)
                    ;
                    ;            (EI)
0126 DB02           IN       R2        ;READ SECTOR ADDRESS BYTE
0128 AE             XRA M              ;COMPARE WITH DESIRED SECTOR
0129 47             MOV B,A            ;B =0 FOR OK, =NON-ZERO FOR ERROR
012A 3E21           MVI A,W3STT+W3CCW
012C D303           OUT W3             ;SEND COMMAND TO W3.
                                       ;THIS COMMAND SETS CCW.  (STT
                                       ;BIT MUST ALSO BE A ONE TO AVOID
                                       ;RESETTING STT.)  THE BIT RING PULSE
                                       ;(BRP) FOLLOWING THE SETTING OF CCW
                                       ;WILL START A BIT BY BIT COMPARISON
                                       ;OF THE DATA READ FROM THE DISK WITH
                                       ;THE DATA READ FROM THE CRC REGISTER.
                                       ;(ALTHOUGH THE CPU WILL READ THE
                                       ;COMPLETED 2ND ZERO BYTE AT THE NEXT
                                       ;BRP, THE DISK DRIVE HEAD WILL BEGIN
                                       ;READING THE 1ST CRC BYTE.)
                                       ;
012E 76             HLT                ;WAIT FOR NEXT INTERRUPT.
                    ;
                    ;INTERRUPT(SECOND ZERO BYTE)
                    ;
                    ;            (EI)
012F DB02           IN       R2        ;READ 2ND ZERO BYTE
                                       ;
0131 12             STAX D             ;ZERO2 =0 FOR OK, =NON-ZERO FOR ERROR
0132 13             INX D              ;DE POINTS TO CRCID
                                       ;
0133 76             HLT                ;WAIT FOR NEXT INTERRUPT.
                    ;
                    ;INTERRUPT(CRC BYTE 1)
                    ;
                    ;            (EI)
```

```
0134 3E20              MVI  A,W3STT    ;TURN OFF CCW
                                       ;
0136 D303              OUT      W3     ;SEND COMMAND TO W3
                                       ;STT=1, CCW=0.  CCW IS RESET.
                                       ;AT NEXT BRP BIT-BY-BIT CRC
                                       ;COMPARISON WILL END.
                                       ;
0138 76                HLT             ;WAIT FOR NEXT INTERRUPT.
            ;
            ;INTERRUPT(CRC BYTE 2)
            ;
            ;         (EI)
0139 DB01              IN       R1     ;INTERRUPT CAUSED BY 2ND CRC BYTE
                                       ;
013B E620              ANI R1DER       ;WAS THERE A CRC ERROR?
013D 12                STAX D          ;CRCID =0 FOR OK, =NON-ZERO FOR ERROR
013E 78                MOV A,B
013F B7                ORA A           ;SECTOR OK?
0140 C25201            JNZ RIM         ;NO, TRY AGAIN
0143 76                HLT             ;YES
            ;
            ;INTERRUPT (FIRST GAP BYTE)
            ;
            ;         (EI)
0144 EB                XCHG            ;HL POINTS TO CRCID, A=0
0145 B6                ORA M           ;TEST CRCID
0146 2B                DCX H
0147 B6                ORA M           ;TEST ZERO2
0148 2B                DCX H
0149 B6                ORA M           ;TEST ZERO1
014A 76                HLT
            ;
            ;INTERRUPT (2ND GAP BYTE)
            ;
            ;         (EI)
014B 2B                DCX H
014C B6                ORA M           ;TEST WTRK. (IS TRACK ADDRESS
                                       ;READ EQUAL TO SOFTWARE TRACK
                                       ;POINTER?)
014D C25201            JNZ RIM         ;ONE OF THE ABOVE IN ERROR, TRY AGAIN.
0150 76                HLT
            ;
            ;INTERRUPT (3RD GAP BYTE)
            ;
            ;         (EI)
0151 C9                RET             ;NORMAL RETURN, ZERO FLAG=1
            ;
            ;ERROR
            ;
0152 DB00     RIM:     IN       R0     ;READ STATUS
0154 E602              ANI R0IRQ       ;WAS INTERRUPT AN INDEX REQUEST?
0156 CA0201            JZ       RIA    ;NO: WAIT FOR NEXT MARK
0159 D306              OUT W6          ;YES, IRQ RESET
015B 0D                DCR C           ;DECREMENT LIMIT
```

```
015C  C20201              JNZ    RIA          ;WAIT FOR NEXT MARK IF NOT 3RD COMPLETE
                                               ;REVOLUTION OF DISK WITHOUT SUCCESS
                                               ;
015F  AF                  XRA    A            ;QUIT
0160  D303                OUT    W3           ;RESET STT
                                               ;
                                               ;
0162  3E01                MVI    A,1
0164  320904              STA    NOGO         ;COULD NOT FIND REQUESTED ID
0167  B7                  ORA    A            ;CLEAR ZERO FLAG
0168  C9                  RET                 ;ERROR RETURN
                  ;
                  ;  READ DATA RECORD
                  ;
                  ;REGISTERS: A,F,B,C,DE,HL
                  ;
                  ;   CALL READ ID RECORD FIRST
                  ;
0169  CDFA00   READ:      CALL   RID          ;READ ID
016C  C0                  RNZ                 ;ERROR IN RID, RETURN
016D  3E60                MVI    A,W3WCS+W3STT
016F  D303                OUT W3              ;SET WRITE CLOCK, LEAVE STT SET
0171  0609                MVI    B,9          ;PASS GAP BYTES 4-12
0173  76       RGAP:      HLT
                  ;
                  ;INTERRUPT (GAP BYTE 4-12)
                  ;
                  ;           (EI)
0174  05                  DCR    B
0175  C27301              JNZ    RGAP
0178  76                  HLT                 ;WAIT FOR GAP BYTE 13.  HEAD HAS
                                               ;NOW PASSED AREA IN GAP THAT CONTAINS
                                               ;UNKNOWN INFORMATION GENERATED WHEN
                                               ;WRITE CURRENT WAS TURNED ON TO WRITE
                                               ;DATA RECORD.
                  ;
                  ;INTERRUPT (GAP BYTE 13)
                  ;
                  ;           (EI)
0179  3E40                MVI    A,W3WCS      ;RESET STT, SET WRITE CLOCK TO
017B  D303                OUT W3              ;PREVENT INTERRUPTS UNTIL FOLLOWING
                                               ;IS DONE.
                                               ;
017D  218004              LXI    H,BUFFR      ;SET HL TO 1ST ADDRESS OF
                                               ;STORAGE BUFFER.
                                               ;
0180  1621                MVI    D,W3STT+W3CCW ;STORE COMMAND TO SET CCW IN
                                               ;D REGISTER.
                                               ;
0182  0EFB                MVI        C,0FBH   ;STORE DATA ADDRESS MARK CODE
                                               ;IN C.
                                               ;
0184  3A0404              LDA    SCTSZ        ;SET SECTOR SIZE
0187  D603                SUI    3
```

49

```
0189 47                    MOV B,A              ;SAVE COUNT IN B
                                                ;
018A 3EA0                  MVI A,W3RCS+W3STT ;SET READ CLOCK, SET STT.
018C D303                  OUT W3
018E 76                    HLT                  ;WAIT FOR ADDRESS MARK.
                    ;
                    ;INTERRUPT (ADDRESS MARK)
                    ;
                    ;          (EI)
018F DB02      .           IN R2                ;READ BYTE
0191 B9                    CMP C                ;IS IT A DATA ADDRESS MARK?
0192 C2C501                JNZ      MARK        ;NO: JUMP TO MARK
0195 76                    HLT                  ;WAIT FOR FIRST DATA BYTE
                    ;
                    ;INTERRUPT (DATA BYTE 1)
                    ;
                    ;          (EI)
0196 DB02                  IN R2                ;YES: READ FIRST DATA BYTE
0198 77                    MOV      M,A         ;STORE FIRST DATA BYTE
                                                ;
0199 23        RLOOP:      INX      H           ;READ AND STORE DATA BYTES
  9A 76                    HLT
                    ;
                    ;INTERRUPT (DATA BYTES 2-(SCTSZ-2) )
                    ;
                    ;          (EI)
019B DB02                  IN R2
019D 77                    MOV      M,A
019E 05                    DCR      B
019F C29901                JNZ      RLOOP
01A2 23                    INX      H
01A3 76                    HLT
                    ;
                    ;INTERRUPT (DATA BYTE #(SCTSZ-1) )
                    ;
                    ;          (EI)
01A4 DB02                  IN R2                ;READ AND STORE DATA BYTE
01A6 77                    MOV      M,A
                                                ;
01A7 7A                    MOV      A,D         ;SET CCW
01A8 D303                  OUT W3
01AA 76                    HLT
                    ;
                    ;INTERRUPT (DATA BYTE #SCTSZ)
                    ;
                    ;          (EI)
01AB 23                    INX H                ;READ AND STORE LAST DATA BYTE
01AC DB02                  IN R2
01AE 77                    MOV      M,A
01AF 76                    HLT
                    ;
                    ;INTERRUPT (FIRST CRC BYTE)
                    ;
                    ;          (EI)
```

```
01B0 3E20              MVI A,W3STT
01B2 D303              OUT W3          ;RESET CCW
01B4 76                HLT
             ;
             ;INTERRUPT (2ND CRC BYTE)
             ;
             ;          (EI)
01B5 DB01              IN R1           ;READ STATUS
01B7 47                MOV B,A         ;SAVE STATUS
01B8 AF                XRA A
01B9 D303              OUT W3          ;RESET STT.  (372 GOES TO WRITE
                                       ;CLOCK AUTOMATICALLY.)
                                       ;
01BB 78                MOV A,B         ;RECALL STATUS
                                       ;
01BC E620              ANI R1DER       ;IS THERE A CRC ERROR?
01BE 320E04            STA CRCDR       ;SET CRC DATA RECORD FLAG
01C1 C8                RZ              ;NO, NORMAL RETURN
                                       ;
                                       ;
                                       ;READ RECORD BUT FOUND
01C2 C3DA01            JMP MK030       ;CRC ERROR
                                       ;
01C5 AF        MARK:   XRA     A       ;RESET STT
01C6 D303              OUT W3
01C8 DB02              IN R2           ;READ MARK AGAIN
01CA DEF8              SBI 0F8H        ;IS IT A "DELETED DATA MARK"?
01CC 320F04            STA ILLMK       ;SET ILLEGAL MARK FLAG
01CF C2D601            JNZ MK010       ;ILLEGAL MARK
01D2 3C                INR A           ;DELETED DATA MARK
01D3 C3D701            JMP MK020
01D6 AF        MK010:  XRA A           ;ILLEGAL MARK
01D7 321004    MK020:  STA DELMK       ;SET DELETED DATA MARK FLAG
01DA 211204    MK030:  LXI H,RRTRY     ;CHECK FOR RETRY
01DD 35                DCR M
01DE C26901            JNZ READ        ;TRY AGAIN
01E1 AF                XRA A
01E2 D303              OUT W3          ;RESET STT
01E4 3C                INR A           ;CLEAR ZERO FLAG TO
                                       ;INDICATE AN ERROR CONDITION
01E5 C9                RET
             ;
             ; WRITE DATA RECORD ROUTINE
             ;
             ;REGISTERS: A,F,B,C,DE,HL
             ;
             ;   CALL READ ID RECORD ROUTINE FIRST
             ;
01E6 CDFA00   WRITE:   CALL RID        ;READ ID
01E9 C0                RNZ             ;ERROR IN RID, RETURN
                                       ;
01EA 0606              MVI B,6         ;COUNT 6 MORE INTERRUPTS FROM ID
01EC 76       WGAP:    HLT             ;RECORD.  (HEAD WILL THEN BE
             ;
```

```
                       ;INTERRUPT
                       ;
                       ;          (EI)
01ED 05                         DCR       B          ;READING 10TH BYTE)
01EE C2EC01                     JNZ       WGAP
                                                     ;
01F1 3EB8                       MVI A,W1CBS+W1CBN ;SET CLOCK BITS AND STROBE
01F3 D301                       OUT W1               ;SET WRITE CLOCK LOGIC TO WRITE
                                                     ;ALL CLOCK BITS ("FF" CLOCK BITS)
                                                     ;FOR DATA
                                                     ;
01F5 AF                         XRA       A          ;SET WRITE DATA REGISTER TO 00.
01F6 D302                       OUT W2
01F8 76                         HLT                  ;WAIT FOR INTERRUPT
                       ;
                       ;INTERRUPT
                       ;
                       ;          (EI)               ;(10TH INTERRUPT SINCE ID RECORD.
                                                     ;HEAD IS READING GAP BYTE 11.)
                                                     ;
01F9 3E70                       MVI A,W3WCS+W3STT+W3WES ;WCS, STT, WES = 1
01FB D303                       OUT W3               ;WRITE CURRENT AND WRITE CLOCK
                                                     ;WILL START AT NEXT BRP
01FD DB01                       IN R1                ;READ STATUS
01FF E604                       ANI R1WFT            ;WRITE FAULT?
0201 CA1602                     JZ WR010             ;NO, CONTINUE
0204 3E02                       MVI A,W0WFR
0206 CDC802                     CALL SETW0           ;WRITE FAULT RESET
0209 3E02                       MVI A,W0WFR
020B CDC002                     CALL CLRW0           ;CLEAR RESET BIT
020E 3E01                       MVI A,1
0210 321104                     STA WRITF            ;YES, SET WRITE FAULT FLAG
0213 C3C600                     JMP ERROR
                                                     ;
0216 76           WR010:        HLT                  ;WAIT FOR INTERRUPT
                       ;
                       ;INTERRUPT
                       ;
                       ;          (EI)
0217 76                         HLT                  ;HEAD BEGINS WRITING. WRITES 00
                                                     ;IN GAP BYTE 12 (11TH INTERRUPT
                                                     ;SINCE ID RECORD)
                       ;
                       ;INTERRUPT
                       ;
                       ;          (EI)
0218 76                         HLT                  ;12TH INTERRUPT.  HEAD WRITES 00
                                                     ;IN BYTE 13
                       ;
                       ;INTERRUPT
                       ;
                       ;          (EI)
0219 76                         HLT                  ;13TH INTERRUPT.  HEAD STARTS BYTE 1
                       ;
```

```
           ;INTERRUPT
           ;
           ;           (EI)
021A 218004          LXI  H,BUFFR       ;SET H,L TO START OF WRITE BUFFER
021D 76              HLT                ;(14TH INTERRUPT.  HEAD STARTS BYTE 15)
           ;
           ;INTERRUPT
           ;
           ;           (EI)
021E 06FB            MVI  B,0FBH        ;LOAD DATA MARK IN B
                                        ;(15TH INTERRUPT.  HEAD STARTS BYTE 16)
                                        ;
0220 0E22            MVI C,W3STT+W3CCG   ;STORE SET CCG COMMAND IN C
                                        ;
0222 16B8            MVI D,W1CBS+W1CBN   ;STORE "FF" CLOCK PATTERN
                                        ;COMMAND IN D
                                        ;
0224 1E20            MVI  E,W3STT       ;STORE RESET CCG COMMAND IN E
                                        ;
0226 3E80            MVI A,W1CBS+W1CBD   ;STORE "C7" DATA MARK CLOCK
                                        ;PATTERN COMMAND IN A
                                        ;
0228 76              HLT                ;WAIT FOR INTERRUPT
           ;
           ;INTERRUPT
           ;
           ;           (EI)
0229 D301            OUT  W1            ;SET "C7" DATA MARK CLOCK PATTERN.
                                        ;(16TH INTERRUPT.  HEAD WRITING
                                        ;17TH AND LAST GAP BYTE)
                                        ;
022B 78              MOV  A,B           ;SET "FB" DATA BITS FOR
022C D302            OUT  W2            ;DATA MARK
                                        ;
022E 79              MOV  A,C           ;SET CCG.  THIS CAUSES CRC
022F D303            OUT  W3            ;CALCULATION TO BEGIN AT NEXT BRP.
                                        ;
0231 7A              MOV  A,D           ;GET "FF" DATA BIT CLOCK
                                        ;PATTERN IN A
                                        ;
0232 76              HLT                ;WAIT FOR INTERRUPT.
           ;
           ;INTERRUPT
           ;
           ;           (EI)
0233 D301            OUT  W1            ;SET "FF" DATA BIT CLOCK PATTERN
                                        ;FOR NEXT BYTE.  HEAD NOW BEGINS
                                        ;WRITING DATA MARK
                                        ;
0235 7B              MOV  A,E           ;RESET CCG.  (CCG MUST BE RESET
0236 D303            OUT  W3            ;BEFORE NEXT BRP OR CRC CALCULATION
                                        ;WOULD BEGIN AGAIN.)
                                        ;
0238 7E              MOV  A,M           ;LOAD FIRST DATA BYTE IN
```

53

```
0239 D302          OUT W2
                                          ;
023B 76            HLT                    ;WAIT FOR INTERRUPT
             ;
             ;INTERRUPT
             ;
             ;          (EI)
023C 3A0404        LDA SCTSZ              ;SET SECTOR SIZE
023F 3D            DCR A
0240 47            MOV B,A                ;SAVE COUNT IN B.  HEAD
                                          ;BEGINS WRITING FIRST DATA BYTE.
                                          ;
0241 23     WLOOP: INX    H               ;WRITE DATA BYTES 2 THRU NBSCT
0242 7E            MOV    A,M
0243 D302          OUT W2
0245 76            HLT
             ;
             ;INTERRUPT
             ;
             ;          (EI)
0246 05            DCR    B
0247 C24102        JNZ    WLOOP
                                          ;
024A 3E21          MVI A,W3STT+W3CCW ;SET CCW.  IN WRITE MODE THE 372
024C D303          OUT W3                 ;WILL BEGIN WRITING BITS FROM THE
                                          ;CRC REGISTER AT THE NEXT BRP
                                          ;FOLLOWING THE SETTING OF CCW.
                                          ;(HEAD IS WRITING DATA BYTE 128)
                                          ;
024E 76            HLT                    ;WAIT FOR INTERRUPT
             ;
             ;INTERRUPT
             ;
             ;          (EI)
024F 76            HLT                    ;HEAD STARTS WRITING FIRST CRC BYTE
             ;
             ;INTERRUPT
             ;
             ;          (EI)
0250 3EFF          MVI    A,0FFH          ;LOAD FF GAP BYTE IN WRITE DATA
0252 D302          OUT W2                 ;REGISTER (HEAD BEGINS WRITING
                                          ;2ND CRC BYTE.)
                                          ;
0254 3E20          MVI A,W3STT            ;RESET CCW COMMAND.  CRC BIT
0256 D303          OUT W3                 ;WRITING WILL STOP AT NEXT BRP.
                                          ;
0258 76            HLT                    ;WAIT FOR INTERRUPT.
             ;
             ;INTERRUPT
             ;
             ;          (EI)
0259 3E24          MVI A,W3STT+W3WER ;WRITE ENABLE RESET.  WRITE
025B D303          OUT W3                 ;CURRENT WILL STOP AT NEXT
                                          ;BRP.  (HEAD BEGINS WRITING 1ST
```

```
                                        ;GAP BYTE.)
                                        ;
025D 76                 HLT             ;WAIT FOR INTERRUPT
              ;
              ;INTERRUPT
              ;
              ;       (EI)
025E AF                 XRA     A       ;RESET STT.
025F D303               OUT W3
                                        ;
0261 C9                 RET             ;DATA RECORD IS WRITTEN.
              ;
              ;
              ;
              ;  STEP IN
              ; REGISTERS: A,F,B,DE,HL
              ; SUBROUTINES: INDXA,UNSLC,DELAY,SETW0
0262 CDD002   STI:      CALL UNSLC
0265 06C0               MVI B,W4STS+W4SID
0267 3E01               MVI A,1         ;INCREMENT TKPTR
0269 C37302             JMP ST010
              ;
              ;  STEP OUT
              ; REGISTERS: A,F,B,DE,HL
              ; SUBROUTINES: INDXA,UNSLC,DELAY,CLRW0
026C CDD002   STO:      CALL UNSLC
026F 0680               MVI B,W4STS
0271 3EFF               MVI A,-1        ;DECREMENT TKPTR
              ;
0273          ST010     EQU $
                        IF NOT MU
0273 211404             LXI H,TKPTR
                        ENDIF
                        IF MU
                        LXI D,TKPTR
                        CALL INDXA
                        ENDIF
0276 86                 ADD M           ;INC/DEC TKPTR
0277 77                 MOV M,A
0278 3E2B               MVI A,LHCTK     ;CHECK FOR WRITE CURRENT CHANGE
027A BE                 CMP M           ;A=LHCTK-TKPTR
027B 3E04               MVI A,W0LCT
027D FA8602             JM ST020        ;TRACK>LHCTK
0280 CDC002             CALL CLRW0      ;TRACK<OR=LHCTK,
                                        ;TURN OFF LOW CURRENT
0283 C38902             JMP ST030
0286 CDC802   ST020:    CALL SETW0      ;TRACK>LHCTK,
                                        ;TURN ON LOW CURRENT
0289 78       ST030:    MOV A,B
028A D304               OUT W4          ;SET DIRECTION
028C F620               ORI W4SOS       ;TURN ON SOS
028E D304               OUT W4          ;RISING EDGE OF SOS
0290 E6DF               ANI NOT W4SOS   ;TURN OFF SOS
0292 D304               OUT W4          ;TRAILING EDGE OF SOS
```

```
                        ;    DELAY 10 MSEC
0294 060A               MVI B,10
0296 CDE202             CALL DELAY
0299 C9                 RET
                    ;
                    ;
                    ;  HEAD LOAD SUBROUTINE
                    ;  REGISTERS: A,F,B,DE,HL
                    ;  SUBROUTINES: UNSLC,DELAY,INDXA,SETW0
029A          HDLD      EQU $                ;CHECK HEAD STATUS
                        IF NOT MU
029A 211604             LXI H,HEAD
                        ENDIF
                        IF MU
                        LXI D,HEAD
                        CALL INDXA
                        ENDIF
029D 7E                 MOV A,M
029E B7                 ORA A                ;IS HEAD LOADED ALREADY?
029F C0                 RNZ                  ;YES
02A0 CDD002             CALL UNSLC           ;NO, SELECT UNIT
02A3 3E08               MVI A,W0HLD
02A5 CDC802             CALL SETW0           ;LOAD HEAD
02A8 0628               MVI B,40             ;WAIT 40 MSEC
02AA CDE202             CALL DELAY
02AD 3E01               MVI A,1              ;SET HEAD STATUS
02AF C3BB02             JMP UL010
                    ;
                    ;  UNLOAD HEAD SUBROUTINE
                    ;  REGISTERS: A,F,DE,HL
                    ;  SUBROUTINES: UNSLC,INDXA,CLRW0
02B2 CDD002    UNLD:    CALL UNSLC
02B5 3E08               MVI A,W0HLD
02B7 CDC002             CALL CLRW0           ;UNLOAD HEAD
02BA AF                 XRA A
02BB          UL010     EQU $                ;UPDATE HEAD STATUS
                        IF NOT MU
02BB 211604             LXI H,HEAD
                        ENDIF
                        IF MU
                        LXI D,HEAD
                        CALL INDXA
                        ENDIF
02BE 77                 MOV M,A
02BF C9                 RET
                    ;
                    ;WR0 MANAGER
                    ;   A=BITS TO BE CLEARED/SET
                    ;    REGISTERS: A,F,HL
02C0 211304    CLRW0:   LXI H,WR0
02C3 2F                 CMA
02C4 A6                 ANA M                ;CLEAR
02C5 C3CC02             JMP CR010
02C8 211304    SETW0:   LXI H,WR0
```

56

```
02CB B6                ORA M              ;SET
02CC D300      CR010:  OUT W0
02CE 77                MOV M,A            ;SAVE A COPY OF W0
02CF C9                RET
               ;
               ;
               ; UNIT SELECT SUBROUTINE
               ;   UNIT=UNIT#(1-4)
               ; REGISTERS: A,F
               ;   ERROR RETURN: JMP ERROR
02D0           UNSLC   EQU $
                       IF NOT MU
02D0 3E01              MVI A,1
                       ENDIF
                       IF MU
                       LDA UNIT
                       DCR A             ;A=0-3
                       ANI W4UBA         ;MASK UNIT#
                       ENDIF
02D2 F604              ORI W4UBS         ;TURN ON STROBE
02D4 D304              OUT W4            ;SELECT UNIT
02D6 DB00              IN R0             ;CHECK FOR READY
02D8 E640              ANI R0RYB         ;READY?
02DA C0                RNZ               ;YES
02DB 2F                CMA               ;NO
02DC 320804            STA SLCTF         ;SET FLAG
02DF C3C600            JMP ERROR
               ;
               ;
               ;
               ;  DELAY SUBROUTINE
               ;        B= # OF MSEC  (MAX=256 WITH B=0)
               ;  REGISTERS: A,F,B
               ;
02E2 3E04      DELAY:  MVI A,W6TRR       ;TURN ON TRR
02E4 D306              OUT W6            ;RESET TIMER REQUEST
               ; WAIT FOR TRQ RST STATUS
02E6 DB00      D010:   IN R0             ;READ STATUS
02E8 E604              ANI R0TRQ         ;CHECK FOR TRQ
02EA CAE602            JZ D010           ;WAIT FOR 1 MSEC
02ED 05                DCR B             ;DONE?
02EE C2E202            JNZ DELAY         ;NO
02F1 C9                RET               ;YES
               ;
               ;
               ;
               ;
                       IF MU
               ;INDEXED ADDRESSING SUBROUTINE
               ;   INPUT:DE=BASE
               ;         UNIT=INDEX
               ; OUTPUT:HL=DE+(UNIT-1)
               ; REGISTERS: F,DE,HL
               INDXA:  LXI H,UNIT
```

```
                         MOV L,M
                         MVI H,0
                         DCR L
                         DAD D
                         RET
                         ENDIF
                   ;
                   ;
                   ;
                   ; DISK FORMATTING ROUTINE
                   ;
02F2 CD1F00        FRMAT:  CALL INIT       ;INITIALIZE DISK UNIT
                   ; INITIALIZE ADDRESS POINTERS
02F5 CD9A02                CALL HDLD       ;LOAD HEAD
02F8 210100                LXI H,1         ;H=00=TRACK ADDRESS
                                           ;L=01=SECTOR ADDRESS

                   ;
                   ; SET UP COMMANDS
02FB 3EB8          FM030:  MVI A,W1CBS+W1CBN
02FD D301                  OUT W1          ;SET CLOCK BITS
02FF 3EFF                  MVI A,0FFH
0301 D302                  OUT W2          ;SET WRITE DATA=0FFH
0303 AF                    XRA A
0304 D303                  OUT W3          ;RESET STT
0306 3E78                  MVI A,W3WCS+W3STT+W3WES+W3IXS
0308 D303                  OUT W3          ;SET 372 TO START WRITING
                                           ;AT INDEX HOLE
030A FB                    EI              ;ENABLE INTERRUPTS AND
030B 76                    HLT             ;WAIT FOR INDEX
                   ;
                   ; INTERRUPT (INDEX START)
                   ;
                   ;       (EI)      HEAD IS WRITING FIRST GAP BYTE
030C 3E02                  MVI A,W6IRR
030E D306                  OUT W6          ;RESET INDEX REQUEST
                   ; WRITE PRE-INDEX GAP
0310 0627                  MVI B,39        ; B=NUMBER OF OFFH GAP BYTES
0312 76            FM040:  HLT             ;WAIT FOR BRP INTERRUPT
                   ;
                   ; INTERRUPT (DATA REQUEST)
                   ;
                   ;       (EI)      HEAD WRITES GAP BYTES 2-40
0313 05                    DCR B           ;DONE?
0314 C21203                JNZ FM040       ; NO, REPEAT
0317 AF                    XRA A           ;YES, CHANGE GAP
0318 D302                  OUT W2          ;BYTE TO 00H
031A 0605                  MVI B,5         ;B=BYTE COUNT
031C 76            FM050:  HLT
                   ;
                   ; INTERRUPT
                   ;
                   ;       (EI)      HEAD WRITES GAP BYTES 41-45
031D 05                    DCR B           ;DONE?
031E C21C03                JNZ FM050       ; NO, REPEAT
```

```
0321 76                HLT              ;YES
              ;
              ; INTERRUPT
              ;
              ;      (EI)      HEAD IS WRITING GAP BYTE 46
              ; WRITE INDEX ADDRESS MARK
0322 3E90              MVI A,W1CBS+W1CBI
0324 D301              OUT W1          ;CHANGE CLOCK BITS
0326 3EFC              MVI A,0FCH
0328 D302              OUT W2          ;SET WRITE DATA=0FCH
032A 76                HLT             ;WRITE MARK
              ;
              ; INTERRUPT
              ;
              ;      (EI)      HEAD IS WRITING INDEX ADDRESS MARK
              ; WRITE POST-INDEX GAP
032B 3EB8              MVI A,W1CBS+W1CBN
032D D301              OUT W1          ;SET CLOCK BITS
032F 3EFF              MVI A,0FFH
0331 D302              OUT W2          ;SET WRITE DATA=0FFH
0333 061A              MVI B,26        ; B=BYTE COUNT
0335 76      FM060:    HLT             ;WRITE GAP BYTE
              ;
              ; INTERRUPT
              ;
              ;      (EI)      HEAD WRITES GAP BYTES 1-26
0336 05                DCR B           ;DONE?
0337 C23503            JNZ FM060       ; NO, REPEAT
              ;
033A AF      FM070:    XRA A           ;BEGINNING OF SECTOR WRITE LOOP
                                       ;-EXECUTED 26 TIMES
033B D302              OUT W2          ;SET WRITE DATA=00H
033D 76                HLT
              ;
              ; INTERRUPT
              ;
              ;      (EI)      HEAD IS WRITING 1ST OF 6 00 BYTES
033E 76                HLT
              ;
              ; INTERRUPT
              ;
              ;      (EI)      2ND OF 6
033F 76                HLT
              ;
              ; INTERRUPT
              ;
              ;      (EI)      3RD OF 6
0340 76                HLT
              ;
              ; INTERRUPT
              ;
              ;      (EI)      4TH OF 6
0341 76                HLT
              ;
```

```
                        ; INTERRUPT
                        ;
                        ;       (EI)        5TH OF 6
0342 06FE                       MVI B,0FEH       ; LOAD ID MARK IN B
0344 0E22                       MVI C,W3STT+W3CCG ; STORE SET CCG COMMAND IN C
                                                ; (ALSO RESETS IXS)
0346 16B8                       MVI D,W1CBS+W1CBN ; STORE "FF" CLOCK PATTERN
                                                ;COMMAND IN D
0348 1E20                       MVI E,W3STT       ; STORE RESET CCG COMMAND IN E
034A 3E80                       MVI A,W1CBS+W1CBD ; STORE "C7" DATA MARK CLOCK
                                                ;PATTERN COMMAND IN A
034C 76                         HLT
                        ;
                        ; INTERRUPT
                        ;
                        ;       (EI)        HEAD IS WRITING 6TH OF 6 00 GAP BYTES
034D D301                       OUT W1          ;SET "C7" DATA MARK CLOCK PATTERN.
034F 78                         MOV A,B         ;SET "FE" DATA BITS FOR
0350 D302                       OUT W2          ;ID MARK
0352 79                         MOV A,C         ;SET CCG.  THIS CAUSES CRC
0353 D303                       OUT W3          ;CALCULATION TO BEGIN AT NEXT BRP.
0355 7A                         MOV A,D         ;GET "FF" DATA CLOCK BIT
                                                ;PATTERN IN A
0356 76                         HLT
                        ;
                        ; INTERRUPT
                        ;
                        ;       (EI)        HEAD IS WRITING ID ADDRESS MARK
0357 D301                       OUT W1          ;SET "FF" DATA CLOCK BIT PATTERN
                                                ;FOR NEXT BYTE.  HEAD NOW BEGINS
                                                ;WRITING ID MARK
0359 7B                         MOV A,E         ;RESET CCG.  (CCG MUST BE RESET
035A D303                       OUT W3          ;BEFORE NEXT BRP OR CRC CALCULATION
                                                ;WOULD BEGIN AGAIN.)
035C 7C                         MOV A,H         ;LOAD TRACK ADDRESS
035D D302                       OUT W2
035F 76                         HLT             ;WAIT FOR INTERRUPT
                        ;
                        ; INTERRUPT
                        ;
                        ;       (EI)        HEAD IS WRITING TRACK ADDRESS
0360 AF                         XRA A
0361 D302                       OUT W2          ;SET DATA BYTE=00H
0363 76                         HLT
                        ;
                        ; INTERRUPT
                        ;
                        ;       (EI)        HEAD IS WRITING FIRST ZERO BYTE
0364 7D                         MOV A,L
0365 D302                       OUT W2          ;SET DATA BYTE=SECTOR ADDRESS
0367 76                         HLT
                        ;
                        ; INTERRUPT
                        ;
```

```
;                    (EI)        HEAD IS WRITING SECTOR ADDRESS
0368 AF                  XRA A
0369 D302                OUT W2          ;SET DATA BYTE=00H
036B 76                  HLT
          ;
          ; INTERRUPT
          ;
          ;          (EI)        HEAD IS WRITING 2ND ZERO BYTE
036C 3E21                MVI A,W3STT+W3CCW ; SET CCW. IN WRITE MODE THE 372
036E D303                OUT W3          ;WILL BEGIN WRITING BITS FROM THE
                                         ;CRC REGISTERS AT THE NEXT BRP
                                         ;FOLLOWING THE SETTING OF CCW.
0370 76                  HLT             ;WAIT FOR INTERRUPT
          ;
          ; INTERRUPT
          ;
          ;          (EI)        HEAD IS WRITING FIRST CRC BYTE
0371 76                  HLT
          ;
          ; INTERRUPT
          ;
          ;          (EI)        HEAD IS WRITING 2ND CRC BYTE
0372 3EFF                MVI A,0FFH      ; LOAD FF GAP BYTE IN WRITE DATA
0374 D302                OUT W2          ;REGISTER
0376 3E20                MVI A,W3STT     ; RESET CCW COMMAND.  CRC BIT
0378 D303                OUT W3          ;WRITING WILL STOP AT NEXT BRP.
037A 060B                MVI B,11        ; B=BYTE COUNT
037C 76      FM080:      HLT
          ;
          ; INTERRUPT
          ;
          ;          (EI)        HEAD WRITES GAP BYTES 1-11
037D 05                  DCR B           ;DONE?
037E C27C03              JNZ FM080       ; NO, REPEAT
0381 AF                  XRA A           ;YES, CHANGE GAP BYTE
0382 D302                OUT W2          ;TO 00H
0384 76                  HLT
          ;          (EI)        BYTE 12
0385 76                  HLT
          ;          (EI)        BYTE 13
0386 76                  HLT
          ;          (EI)        BYTE 14
0387 76                  HLT
          ;          (EI)        BYTE 15
0388 76                  HLT
          ;
          ; INTERRUPT
          ;
          ;          (EI)        HEAD IS WRITING GAP BYTE 16
0389 06FB                MVI B,0FBH      ; LOAD DATA MARK IN B
038B 0E22                MVI C,W3STT+W3CCG ; STORE SET CCG COMMAND IN C
038D 16B8                MVI D,W1CBS+W1CBN ; STORE "FF" CLOCK PATTERN
                                          ;COMMAND IN D
038F 1E20                MVI E,W3STT     ; STORE RESET CCG COMMAND IN E
```

61

```
0391 3E80              MVI A,W1CBS+W1CBD ; STORE "C7" DATA MARK CLOCK
                                        ;PATTERN COMMAND IN A
0393 76          HLT                    ;WAIT FOR INTERRUPT
                 ;
                 ; INTERRUPT
                 ;
                 ;      (EI)       HEAD IS WRITING GAP BYTE 17
0394 D301        OUT W1                 ;SET "C7" DATA MARK CLOCK PATTERN.
0396 78          MOV A,B                ;SET "FB" DATA BITS FOR
0397 D302        OUT W2                 ;DATA MARK
0399 79          MOV A,C                ;SET CCG.  THIS CAUSES CRC
039A D303        OUT W3                 ;CALCULATION TO BEGIN AT NEXT BRP.
039C 7A          MOV A,D                ;GET "FF" DATA BIT CLOCK
                                        ;PATTERN IN A
039D 76          HLT                    ;WAIT FOR INTERRUPT.
                 ;
                 ; INTERRUPT
                 ;
                 ;      (EI)       HEAD IS WRITING DATA ADDRESS MARK
039E D301        OUT W1                 ;SET "FF" DATA BIT CLOCK PATTERN
                                        ;FOR NEXT BYTE.
03A0 7B          MOV A,E                ;RESET CCG.  (CCG MUST BE RESET
03A1 D303        OUT W3                 ;BEFORE NEXT BRP OR CRC CALCULATION
                                        ;WOULD BEGIN AGAIN.)
03A3 3EE5        MVI A,0E5H             ;LOAD DATA
03A5 D302        OUT W2
03A7 76          HLT
                 ;
                 ; INTERRUPT
                 ;
                 ;      (EI)        DATA BYTE 1
03A8 067F        MVI B,NBSCT-1
03AA 76     FM100:  HLT
                 ;
                 ; INTERRUPT
                 ;
                 ;      (EI)        HEAD WRITES DATA BYTES 2-NBSCT
03AB 05          DCR B
03AC C2AA03      JNZ FM100
                 ;
03AF 3E21        MVI A,W3STT+W3CCW ; SET CCW.  IN WRITE MODE THE 372
03B1 D303        OUT W3                 ;WILL BEGIN WRITING BITS FROM THE
                                        ;CRC REGISTERS AT THE NEXT BRP
                                        ;FOLLOWING THE NEXT SETTING OF CCW.
03B3 76          HLT
                 ;
                 ; INTERRUPT
                 ;
                 ;      (EI)        HEAD IS WRITING FIRST CRC BYTE
03B4 76          HLT
                 ;
                 ; INTERRUPT
                 ;
                 ;      (EI)        HEAD IS WRITING 2ND CRC BYTE
```

62

```
03B5  3EFF              MVI  A,0FFH          ; LOAD FF GAP BYTE IN WRITE DATA
03B7  D302              OUT  W2              ;REGISTER
03B9  3E20              MVI  A,W3STT         ; RESET CCW COMMAND.  CRC BIT WRITING
03BB  D303              OUT  W3              ; ENDS
03BD  061B              MVI  B,27            ; B=BYTE COUNT
03BF  76        FM110:  HLT
                ;
                ; INTERRUPT
                ;
                ;           (EI)        HEAD WRITES GAP BYTES 1-27
03C0  05                DCR  B               ;DONE?
03C1  C2BF03            JNZ  FM110           ; NO, REPEAT
                ;
                ;
03C4  2C                INR  L               ;INCREMENT SECTOR ADDRESS
03C5  3E1A              MVI  A,26
03C7  BD                CMP  L               ;LAST SECTOR?
03C8  F23A03            JP   FM070           ; NO, WRITE ANOTHER SECTOR
                ;
                ; WRITE FF'S TO END OF TRACK
03CB  76        FM120:  HLT
                ;
                ; INTERRUPT
                ;
                ;           (EI)        HEAD WRITES GAP BYTES 28 TO 247
03CC  DB00              IN   R0              ;READ STATUS
03CE  E602              ANI  R0IRQ           ; INDEX REQUEST?
03D0  CACB03            JZ   FM120           ; NO, CONTINUE
                ;
                ; END OF TRACK
03D3  3E04              MVI  A,W3WER
03D5  D303              OUT  W3              ;WRITE ENABLE AND STT RESET.
                                             ;INDEX REQUEST IS AUTOMATICALLY RESET
                                             ;BY STT RESET.
03D7  3E4C              MVI  A,NTRKS-1
03D9  BC                CMP  H               ;LAST TRACK?
03DA  C8                RZ                   ; YES, FORMATTING COMPLETE
03DB  2E01              MVI  L,1             ; NO, RESET SECTOR ADDRESS
03DD  24                INR  H               ;INCREMENT TRACK ADDRESS
03DE  E5                PUSH H               ;SAVE HL
03DF  0602              MVI  B,02            ;WAIT FOR TUNNEL ERASE HEAD
03E1  CDE202            CALL DELAY           ;TO REACH END OF TRACK BEFORE
03E4  CD6202            CALL STI             ;STEPPING HEAD.
03E7  E1                POP  H               ;RESTORE HL
03E8  C3FB02            JMP  FM030           ; CONTINUE
                ;
03EB              ORG  0400H
                ;COMMAND
0400      CMND:  DS   1                      ;COMMAND(1-NCMDS)
                ;
                ; PARAMETERS
0401      UNIT:  DS   1                      ;FDD UNIT BEING COMMANDED
0402      TRACK: DS   1                      ;TRACK DESIRED
0403      SECTR: DS   1                      ;SECTOR DESIRED
```

63

```
0404            SCTSZ:  DS 1                 ;SECTOR SIZE
0004            NP      EQU $-UNIT           ;NO. OF PARAMETERS
                ;
                ; FLAGS
0405            MERF:   DS 1                 ;MASTER ERROR FLAG
0406            CMDER:  DS 1                 ;COMMAND ERROR FLAG
0407            PRMER:  DS 1                 ;PARAMETER ERROR FLAG
0408            SLCTF:  DS 1                 ;SELECT FAULT
0409            NOGO:   DS 1                 ;FAILED TO FIND SECTOR FLAG
040A            WTRK:   DS 1                 ;WRONG TRACK FLAG
040B            ZERO1:  DS 1                 ;ZERO BYTE 1 NOT ZERO FLAG
040C            ZERO2:  DS 1                 ;ZERO BYTE 2 NOT ZERO FLAG
040D            CRCID:  DS 1                 ;CRC ERROR IN ID FLAG
040E            CRCDR:  DS 1                 ;CRC ERROR IN DATA READ FLAG
040F            ILLMK:  DS 1                 ;ILLEGAL DATA MARK FLAG
0410            DELMK:  DS 1                 ;DELETED DATA MARK FLAG
0411            WRITF:  DS 1                 ;WRITE FAULT FLAG
000D            NF      EQU $-MERF           ;NUMBER OF FLAGS
                ;
                ;COUNTERS, POINTERS, STATUSES
0412            RRTRY:  DS 1                 ;READ RETRY COUNTER
0413            WR0:    DS NU                ;COPIES OF LATEST W0
0414            TKPTR:  DS NU                ;TRACK POINTER FOR EACH UNIT
0415            REVS:   DS NU                ;ELAPSED IDLE REVOLUTIONS
0416            HEAD:   DS NU                ;HEAD STATUS(1=LOADED, 0=UNLOADED)
0017            NB      EQU $-CMND           ;NO. OF BYTES IN DATA AREA
                ;
                ;STACK
                ;
0417                    ORG 480H
                ;
                ; DATA BUFFER
0480            BUFFR:  DS NBSCT
                ;
                ;
0000                    END
```

ABSOLUTE MAXIMUM RATINGS

Ta = 25 C   All voltages measured with respect to VSS

Symbol	Parameter	Min	Max	Unit	Conditions
VDD	VDD Supply Voltage	-1	+16	V	VBB=-5V±5%
VCC	VCC Supply Voltage	-1	+8	V	VBB=-5V±5%
VBB	VBB Supply Voltage	-10	0	V	
VI	Input Voltage	-1	+8	V	VBB=-5V±5%
VO	Output Voltage	-1	+8	V	VBB=-5V±5%
VO	Clock Voltage	-1	+16	V	VBB=-5V±5%
Topt	Operating Free-Air Temp.Range	0	+70	C	
Tstg	Storage Temperature	-40	+125	C	

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.

DC CHARACTERISTICS

Ta = 0-70 Ç   VDD = +12V±5%   VCC = +5V±5%   VBB = -5V±5%   VSS = 0V

Symbol	Parameter		Min	Typ	Max	Unit	Conditions
VIH	Input High Voltage		+3.0		VCC	V	
VIL	Input Low Voltage		0		+0.8	V	
VOH	Output High Voltage		+3.5			V	IOH = -1mA
VOL1	Output Low	CKS REQ, UA0 UA1 UB0 UB1 DB0-DB7			+0.5	V	IOL = +1.7mA
VOL2	Voltage	WD HLD LCT WE WFR SOS SID			+0.5	V	IOL = +3.3mA
VØH	Clock Input High Voltage		+9		VDD	V	
VØL	Clock Input Low Voltage		0		+0.8	V	
ILIH	Input Leakage Current				+10	uA	VI = +3.0V
ILIL	Input Leakage Current				-10	uA	VI = +0.8V
ILØH	Clock Input Leakage Current				+10	uA	VØ = +9.0V
ILØL	Clock Input Leakage Current				-10	uA	VØ = +0.8V
ILOH	Output Leakage Current				+10	uA	VO = +3.5V
ILOL	Output Leakage Current				-10	uA	VO = +0.5V
IDD	Power Supply Current(VDD)			+20		mA	
ICC	Power Supply Current(VCC)			+23		mA	
IBB	Power Supply Current (VBB)				-2	mA	

AC CHARACTERISTICS

Ta = 0-70 C   VDD = +12V±5%   VCC = +5V±5%   VBB = -5V±5%   VSS = CU

Symbol	Parameter		Min	Typ	Max	Unit	Conditions
tcy	Clock Period		480		2000	ns	
tr.tf	Clock Rise & Fall Times		0		50	ns	
tØ1	Ø1 Pulse Width		60			ns	
tØ2	Ø2 Pulse Width		90			ns	
tD1	Ø1 to Ø2 Delay		0			ns	
tD2	Ø2 to Ø1 Delay		70			ns	
tD3	Delay Ø1 to Ø2 Lead.Edges		100			ns	
tOD1	Data Out Delay from Ø1				90	ns	1TTL & CL=30pF
TOD2	Data Out Delay from Ø1	CKS, UA0, UA1 REQ, UB0, UB1			200	ns	1TTL & CL=30pF
		HLD LCT WFR WE SOS SID			200	ns	2TTL & CL=50pF
tOD3	WD Delay Time				120	ns	2TTL & CL=50pF
tOD4	Data Out Delay - DS W/R RS2				200	ns	
tIS1	Data Setup Time to Ø1		150			ns	
tIS2	Data Setup Time to Ø2		120			ns	
tIH1	Data Hold Time from Ø1		10			ns	
tIH2	Data Hold Time from Ø2		10			ns	
tWD	WD pulse width		tD3-40	tD3		ns	
tw	Input Signal Pulse Width *		tcy+160			ns	

* IDX, RYA, RYB, RST, WFT, TØØ, WCK, RCK
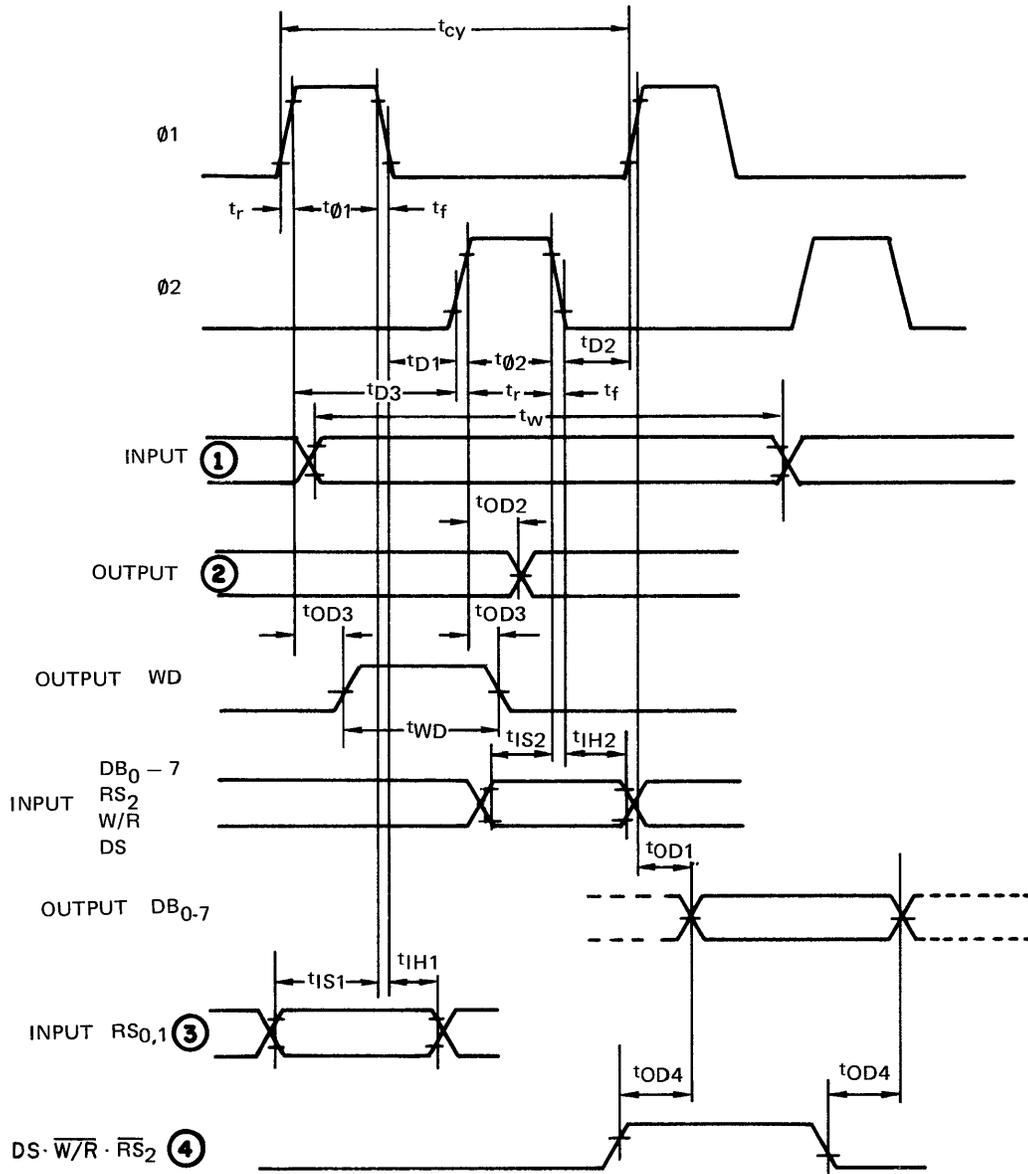
Notes: ① IDX, RYA, RYB, RST, WFT, T∅∅, WCK, RCK.

② CKS, WFR, SOS, SID, REQ, HLD, $UA_{\emptyset,1}$, $UB_{\emptyset,1}$, WE, LCT.

③ $RS_\emptyset$, RS_1 input must not make level transition within t_{IS1} and t_{IH1} times, or register contents may be modified.

④ The logic condition which places μPD372 information on $DB_{\emptyset-7}$ is $DS \cdot \overline{W/R} \cdot \overline{RS_2}$. Care must be taken to insure that this condition is not met inadvertently if DS, W/R and RS_2 are allowed to change state asynchronously.

FIGURE 12
TIMING DIAGRAM
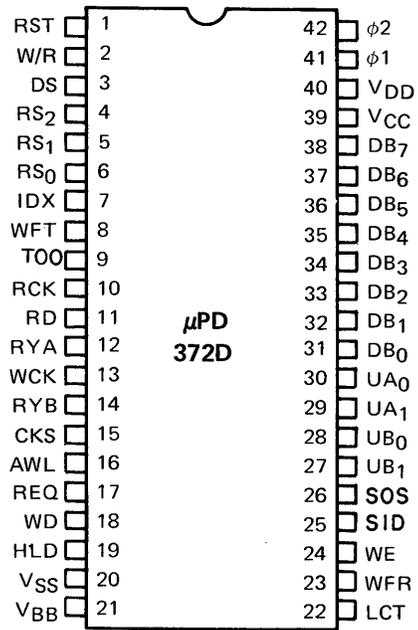
FIGURE 13
PIN CONFIGURATION

Pin assignments:

Pin	Signal		Pin	Signal
1	RST		42	$\phi2$
2	W/R		41	$\phi1$
3	DS		40	V_{DD}
4	RS_2		39	V_{CC}
5	RS_1		38	DB_7
6	RS_0		37	DB_6
7	IDX		36	DB_5
8	WFT		35	DB_4
9	TOO		34	DB_3
10	RCK		33	DB_2
11	RD		32	DB_1
12	RYA		31	DB_0
13	WCK		30	UA_0
14	RYB		29	UA_1
15	CKS		28	UB_0
16	AWL		27	UB_1
17	REQ		26	SOS
18	WD		25	SID
19	HLD		24	WE
20	V_{SS}		23	WFR
21	V_{BB}		22	LCT

(Center: µPD 372D)
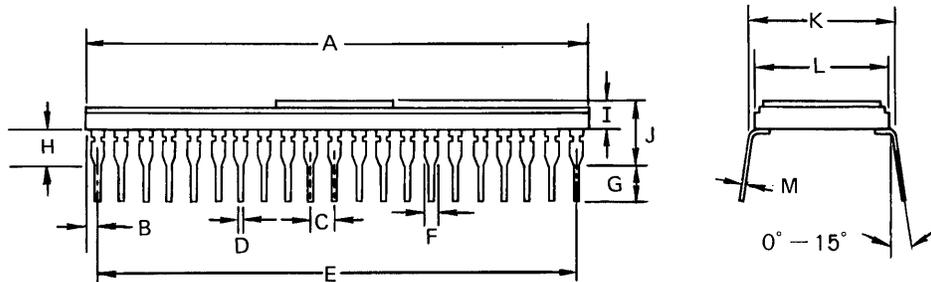
ITEM	MILLIMETERS	INCHES
A	53.5 Max.	2.1 Max.
B	1.35	0.05
C	2.54	0.10
D	0.5	0.02
E	50.80	2.0
F	1.27	0.05
G	2.54 Max.	0.10 Min.
H	1.0 Min.	0.04 Min.
I	4.2 Max.	0.17 Max.
J	5.2 Max.	0.21 Max.
K	15.24	0.60
L	13.50	0.53
M	0.3	0.012

FIGURE 14
PACKAGE OUTLINE