K-1013 DOUBLE DENSITY DISK CONTROLLER

FOR KIM/MTU BUS SYSTEMS

DOUBLE DENSITY FLOPPY DISK CONTROLLER
16 K BYTE MEMORY
256 BYTE BOOTSTRAP ROM
4 DRIVE CAPACITY

MAY, 1980 REV A

MICRO TECHNOLOGY UNLIMITED
P. O. BOX 12106
2806 HILLSBORO STREET
RALEIGH, NC 27605 USA

TABLE OF CONTENTS

K-1013 UNPACKING AND INSTALLATION

The K-1013 Floppy Disk Controller for KIM/MTU bus systems is a carefully engineered, manufactured, and tested product that should operate perfectly when handled and installed according to the following instructions. Note that the board is shipped in a black conductive plastic bag. Since MOS integrated circuits are used, damage from static discharge is possible. It is helpful to reduce static by working in an area with concrete floors and a reasonable humidity level. If this is impossible, at least avoid wearing rubber soled shoes and move slowly in the work area. When unpacking or handling the board, touch the screw sticking up in the middle of the heatsink first and release it last. Note that the preceeding comments apply equally to the user's computer board which of course contains MOS IC's also.

ADDRESS AND FUNCTION SELECT JUMPERS

Due to the variety of systems which may use this board, the K-1013 Floppy Disk Controller has a number of address and function select jumpers. However the board is shipped with a "standard" jumper configuration installed which should be suitable for running the CODOS Disk Operating System in an AIM-65 based system. The characteristics of this standard configuration are listed below:

1. System RAM addresses - 8000-9FFF
2. User RAM addresses - 4000-5FFF
3. Type of disk - 8 inch standard size Shugart compatible single sided drive
4. Reset vector - Goes to KIM-1 monitor (pertains only to KIM-1 based systems)
5. Floppy disk controller interrupt - disabled
6. Option bit in Hardware Status Register - 0
7. Write precompensation - 125NS

It is desirable that initial tests of the board in the user's system be done with these jumper options intact. This should be possible in KIM-1 and AIM-65 based systems. In SYM-1 systems the disk controller System RAM will interfere with the SYM-1 monitor and therefore it must be moved. The best place to move it is to addresses 6000-7FFF (CODOS for the SYM-1 will expect it there as well). If it is moved, the diagnostic program listed in this manual will have to be modified. See modification instructions in the Memory Diagnostic Program listing on page 41.

CONNECTION TO USER'S SYSTEM

As shipped the board requires a source of +8 (+7 to +12) volts unregulated input and a source of +16 (+14 to +20) volts unregulated for power. It is preferable to use the on-board regulators but if only regulated power is available, the regulators may be bypassed. Solder a jumper wire between the two outside pins of VR2 for regulated +5 input. Solder a jumper wire between the two outside pins of VR1 for regulated +12 input. Current drain from +5 is 600MA and from +12 is 125MA.

The easiest method of connection to a KIM-1, SYM-1, or AIM-65 is to use an MTU K-1005 series motherboard/card file. With the card file one simply plugs the processor board into the top slot and the disk controller into one of the other slots. The K-1013 may also be connected directly in parallel with the expansion connector of the processor (except for those signals marked with an * on page 39). If direct connection is used, the wires should not be longer than 4 inches and the ground wire should connect directly from the processor to the K-1013 and be 16 guage or heavier.

INITIAL BOARD TEST

Initial testing of the K-1013 should be performed without a drive connected. This will verify that the board operates properly as a memory, that the disk controller chip is alive, and that the bootstrap ROM, if present, can be read correctly. Perform the following steps to verify that the board is in reasonable operating condition:

1. Inspect the board for shipping damage such as broken parts or bent-over component leads shorting PC traces.

2. After making sure that power is turned off, that the user's power supply has discharged all residual voltages, and that there are no addressing conflicts between the on-board memory and the rest of the system, plug the board into the system and turn the power on.

3. Using the standard system monitor, examine memory location 4000 (or the first User RAM location if the K-1013 jumpers have been changed) several times. The contents will probably be either 00 or FF and each examine operation should report the same value. If the results are different each time it is examined, either the board is not responding to its address or the board has failed to synchronize to the user's system clock which must be 1.0MHz. If the contents are always the same as the high byte of the address (try 4100 and 4200 as well), the board is definitely not responding to its address. Check that the address jumpers are set the way you expect. If the condition persists, refer to the section on troubleshooting.

4. Using the standard system monitor, store something in location 4000 and verify that it can be read back. Repeat for 5000 and a couple of others in the 4000-5FFF range.

5. Repeat step 3 at address 8000 (or the first System RAM location if the K-1013 jumpers have been changed).

6. Since System RAM is write protected after a system reset, you will have to write 00 into the Hardware Control Register before executing step 4 on the system RAM. Simply store 00 into location 9FE8 (or SYSRAM+1FE8) with the monitor. SYM and AIM monitors will report a memory failure since this is a write-only register. The KIM monitor will not write correctly into a write-only register so the following program will have to be put into location 0000 and executed: A9 00 8D E8 9F 4C 22 1C. After the write protect has been turned off, try writing into the system RAM using the procedure in step 4.

7. Examine location 9FEE (or SYSRAM+1FEE) which is the main status register of the disk controller chip. It should read 80. Any other reading indicates a problem with the disk controller chip or board addressing.

8. Examine location 9FE8 (or SYSRAM+1FE8) which is the Hardware Status Register. Bit 7 of the data read should be a one and bit 6 should be a zero while the other bits are undefined. This indicates that no interrupt is pending from the disk controller and that the option bit is 0.

9. Examine location 9F00 (or SYSRAM+1F00) which is the first location in the bootstrap ROM. If the ROM is installed, it should have a value corresponding to the first byte of the bootstrap program (see the bootstrap program listing in the CODOS manual). Otherwise it will be undefined.

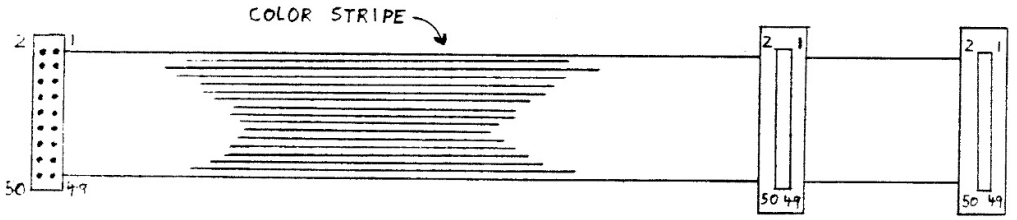
10. If desired, enter the memory diagnostic program on pages 41-44 and execute it to certify that the on-board memory is working properly. Then continue to the next page after turning the power off.

DISK DRIVE CONNECTION

A complete disk system requires a disk drive, connecting cable, and power supply as well as the K-1013 disk controller and the user's computer. Since the K-1013 disk controller comes without drives, cable, or power supply, there is a variety of possible configurations.

Disk drives with a Shugart compatible connector are preferred. These include but are not limited to: Shugart SA-800, Siemans FD-400, and Qume Datatrak 8. Either the single-sided or double-sided versions of the above drives may be used however for double-sided applications the Qume Datatrack 8 is recsmmended. If a non Shugart compatible drive is used, see the section entitled Disk Drive Characteristics.

Assuming that a Shugart compatible disk is used, the cable required is extremely simple. It should be a 50 wire ribbon cable with .050" between conductor centers (3M type number 3365-50) with a female ribbon cable connector on one end (mates with dual row .025" square posts on a .1" grid, 3M type number 3425-0000), and a female card edge connector on the other end (mates with 50 pin double-sided PC edge fingers on .1" spacing, 3M type number 3415-0001). If two or more drives are to be used in a system, additional female card edge connectors can be pressed onto the same cable with a spacing of 6" between centers (assuming upright mounting of the drives with not more than 1" spacing between drives) up to a total of 4 connectors. The total cable length should not exceed 10 feet and lengths of less than 5 feet are preferred. A drawing of the required cable configuration is shown below:



If more than one disk drive is used, the drive select jumpers in the drives will have to be re-configured. Following the instructions in the drive manual, set the jumpers in the first drive for 0, the next for 1, etc. The disk controller does NOT use binary select so ignore anything in the drive manual about binary select.

Most disk drives have addioional jumpers to control other aspects of their operation. The list below shows what the K-1013 expects of the drive and the jumpers should be configured accordingly.

1. The READY line should be daisy chained, not radial.

2. The stepper motor should either be energized continuously or when step pulses are present. Energizing only when the head is loaded or the drive is selected is NOT acceptable.

3. The Track 0 Sense should be daisy chained, not radial.

4. If an activity LED is on the drive front panel, it is of most value if it turns on when the drive's head is loaded.

5. The controller board has its own data separator. Any separator in the drive should be disabled and the Raw Read Data should appear on pin 46.

6. The controller board uses soft sectoring. Any sector separator circuitry on the drive should be disabled and the index sense signal should appear on pin 20.

7. If present, the Write Protect Sensor should be enabled.

8. If present on a double sided drive, the Double Sided Sensor should be enabled.

9. If a door lock solenoid is provided, it probably should not be used unless the disk is part of a turnkey system with unsophisticated operators. If it is used, connect it to the Head Load signal so that the door is locked only when the head is loaded.

10. Any other radial vs daisy chain options should be set for daisy chain.

11. If in doubt about any other options, remember that the disk controller constantly scans the disk drive status at about a 1kHz rate (except when actually reading or writing) by selecting the drives one-at-a-time. Thus Drive Select alone should not initiate functions such as head load, stepper power, or light the front panel activity indicator.

INITIAL DISK TEST

The following tests verify that the disk drive operates correctly and that communication between the controller board and disk drive functions properly. (Note: If the customer is certain that the disk drive is in good operating condition and that the cabling is correct (such as from previous experience), this initial disk test may be skipped and the CODOS manual consulted for CODOS installation procedures.)

1. With the disk drive (or drives) connected to the K-1013 and powered up, apply power to the computer and enter the Level 1 Disk Test Program listed on page 45. Be sure to put zeroes in locations 0009 and 000A for the initial test. Later, the drive number can be put into location 0009 to test other dirves on the system and the side number can be put into location 000A to test both sides of a double-sided disk drive, if used.

2. Insert a blank disk (not the APEX-65 distribution disk!) into drive 0 and start execution at location 4026 which is the random seek test for drive 0. It will first seek to track zero using the Recalibrate command and then do 256 random seeks followed by a seek to track 0. It will then halt with a BRK instruction. Examine location 0000 which will contain 00 if the test executed successfully. Otherwise it will contain an error code which can be used to track down the cause of the difficulty. If additional drives are installed, repeat the test for them by storing the drive number in location 0009. (Note that a 10MS seek speed is used. If the disk drive is rated for faster seeking, the test may be repeated at the faster speed by changing location 4001 to 8F for 8MS, AF for 6MS, or DF for 3MS.)

3. Enter the Level 2 Disk Test Program (be sure to keep Level 1 in memory since subroutines in it are used by Level 2). Insert a blank disk into drive 0 and start execution at 415C. The disk will be formatted in CODOS format (that means absolutely erased!). When the program halts with a BRK, examine the error code at location 0000 which should be 10.

4. Start execution at 422E. This routine reads all sectors on the formatted disk and will stop on any kind of error. It should run indefinitely with a good quality diskette. If it stops, examine location 0000 which will contain an error code. If a Read Error is indicated (code=23), examine the controller status bytes in locations 0001, 0002, and 0003 to determine the kind of read error (see uPD 765 status bytes description on page 31). Then look at the track number in location 000B and sector number in location 000C. If they are both zero then no sectors have been read and there is a problem in the disk drive or cable and the Troubleshooting section should be consulted. An occasional read error is permissable, particularly on the inside tracks. Typical error rates with disk drives and diskettes rated for double-density operation are one per hour.

5. At this point it Is safe to go ahead and install the CODOS Disk Operating System (see the CODOS manual). If desired however, Level 3 of the Disk Test Program may be entered into memory. Start execution at location 4304. The program will first write random data on each sector of the disk and then read it back and check for accuracy against the same random byte stream. The program should stop after about 11 minutes with 30 in location 0000. Anything else indicates an error condition.

PROGRAMMING

Thorough understanding of the following section is not necessary unless the customer wishes to write his own disk handling code. Normally the CODOS Disk Operating System supplied with the K-1013 board performs all of the disk handling functions required. However, special high speed data acquisition applications or utility programs for translating other diskette sector formats into CODOS standard format will require direct programming of the K-1013.

The following discussion is just a summary of the information needed for successful disk hardware programming. Its purpose is to establish an overview and point out the pitfalls discovered during development of the K-1013 and CODOS. Answers to detailed programming questions can be found in the uPD765 Controller Chip Data Sheet on page 27.

K-1013 MEMORY MAP

The K-1013 looks to the using system like two completely independent 8K blocks of memory. Each block may be independently addressed on any 4K boundary. The User RAM block is totally uncommitted in its use and can simply be regarded as an 8K bonus. Note however that the disk controller IC has direct memory access to this block whereas data must be moved by program loops to other memory boards in the user's system.

The 8K of addresses assigned to the System RAM block also includes the I/O registers for the uPD765 chip and the bootstrap ROM. If the beginning of the system RAM is called SYSRAM, then the following chart gives the addresses used in the system RAM:

SYSRAM to SYSRAM+1EFF	Read/write memory; can be software write protected.
SYSRAM+1F00 to SYSRAM+1FE7	Bootstrap loader PROM
SYSRAM+1FE8	Read - Hardware Status Read
	Write - Hardware Control Write
SYSRAM+1FEA	Write-only Set DMA Address
SYSRAM+1FEE	Read-only uPD765 Main Status Register
SYSRAM+1FEF	Read/Write uPD765 Data Register
SYSRAM+1FF0 to SYSRAM+1FFF	Remainder of bootstrap loader PROM

Bootstrap Loader PROM

The Bootstrap Loader PROM is a 256 byte fusible link PROM selected for its small size and low cost compared with erasible PROM's. Eight of the 256 bytes are not available because they are overlayed by I/O register addresses. This PROM is normally supplied by MTU already programmed with a bootstrap loader that will read the CODOS Disk Operating System from disk and jump to it.

Hardware Status Read

Only two bits of the data read from this address are significant. Bit 7, which can be tested with the BMI and BPL 6502 instructions, is a zero if the uPD765 is requesting an interrupt and is a one if not. During DMA operations, this bit must be tested to determine when the operation is complete rather than reading the uPD765 main status register. Bit 6 is connected to the Option Jumper on the K-1013 and is a one if the jumper is installed. This is a read-only register.

Hardware Control Write

Only two bits of the data written to this register are significant. Bit 0 sets the DMA data transfer direction. When set to zero, data flows from K-1013 on-board memory to the disk and when set to one, data flows from the disk to K-1013 on-board memory. This bit must always agree with the read or write command given to the uPD765 for correct data transfer. System reset forces this bit to a zero.

Bit 1 controls write protect of the K-1013 System RAM. When it is zero, normal write operation is allowed. When it is a one, writing into the system RAM by the 6502 is prevented. The setting of this bit has no effect on DMA operations however. System reset forces this bit to a one.

DMA Address Register

This is a write-only register used to specify where in K-1013 on-board memory a DMA transfer to or from the disk is to start. Only the upper 8 bits of the 14 bit DMA address counter may be set by writing to SYSRAM+1FEA, the lower 6 bits will always be cleared to zero. Bits 0-5 of the DMA Address Register correspond exactly to bits 6-11 of the desired DMA address. Bits 6 and 7 of the DMA Address Register should be set according to the table below:

K-1013 RAM BLOCK	BLOCK STARTS ON EVEN 4K BOUNDARY	HALF OF BLOCK	BIT 7	BIT 6
User	Yes	Lower	0	0
User	Yes	Upper	0	1
User	No	Lower	0	1
User	No	Upper	0	0
System	Yes	Lower	1	0
System	Yes	Upper	1	1
System	No	Lower	1	1
System	No	Upper	1	0

One may write a subroutine, such as in the diagnostic program listing, to accept the desired DMA starting address and set the DMA Address Register accordingly.

Once set, the DMA address register will increment on every DMA cycle performed by the uPD765 disk controller. Thus if disk sectors are to be read into consecutive memory locations, the DMA Address Register need not be set at the beginning of every sector. Please note however that if the 8K block of K-1013 memory being read into starts on an odd 4K boundary (such as 3000, 5000, 7000, etc.) the DMA address register will not cross the 4K boundary in the middle of the block correctly. Thus if User RAM starts at 5000 for example, the DMA Address Register should not be allowed to count from 5FFF to 6000 because it will do so incorrectly. If the block starts on an even 4K boundary, such as 4000, there is no problem in counting from 4FFF to 5000.

uPD765 Registers

The disk controller chip itself has only two registers. The Main Status Register is a read-only register that the uPD765 uses to tell the disk program what it is expecting and a little bit of information about the status of things. The uPD765 data register is used to send commands to the chip and receive detailed status from it. Operation in DMA mode is expected in which case the data register is not used for disk data. Please note that during a Read, Write, or Format command execution (which normally do DMA), that the program should not read or write any of the uPD765 registers. The reason is that the uPD765 cannot recover fast enough after a DMA cycle to do a CPU cycle correctly.

COMMUNICATING WITH THE uPD765

One of the problems in designing a floppy disk controller is the wide variety of data that must be exchanged with the using system. If a different address was used for each type of data, over a dozen addresses would be required. The uPD765 chip used by the K-1013 approaches this problem by providing a single data "port" and then using an internal counter to direct the data to or from the correct internal register. With such a setup it is imperative that the internal counter be synchronized with the data so that it goes to or from the desired internal registers. The Main Status Register in the uPD765 is used to insure synchronization.

Actually the uPD765 tells the using system what should be done next rather than vice-versa. This is accomplished with two bits in the Main Status Register. Bit 7 is called Request For Master (RQM) and when it is a one, the uPD765 wants the using system to do something. If it is a zero, the uPD765 is busy and the using system must not read or write the data register (unless the non-DMA mode is being used which is not considered here). Bit 6 indicates whether the uPD765 wishes to talk or listen. If it is a zero (and RQM is a one), the uPD765 is prepared to receive a command. If it is a one (and RQM is a one), the uPD765 has one or more status bytes available for reading.

Sending Commands to the uPD765

A command is sent to the uPD765 by first making sure that it is prepared to receive one (Main Status Register bit 7=1, bit 6=0). Then the first byte of the command is written into the data register. This action sets the internal address counter which distributes the remainder of the command bytes. It is important to test bit 7 of the Main Status Register for a one before each byte of the command is sent because of uncertain internal delays in disposing of command bytes. Note that the exact number of bytes defined for each command must be sent to the 765; no fewer and no more. The using system must therefore be cognizant of the number of bytes the command requires because the 765 does not signal when the correct number has been received. When all of the command bytes have been received, the uPD765 starts executing the command and will be busy until the requested action is complete. (See the data sheet for details on simultaneous seek commands.)

Receiving Status from the uPD765

When the command is completed (except for Specify) the uPD765 will turn its Interrupt Request on which can be sensed by reading the K-1013 Hardware Status Register bit 7. If the command was not a Specify, Seek, or Recalibrate, the 765 is now ready to send status bytes back to the using system. The program should test the Main Status Register for bit 7=1 and bit 6=1 before reading each status byte and status bytes must be read until bit 6 returns to a zero indicating readiness for the next command. Note that different commands give different numbers of status bytes and that their meaning depends on the command. The uPD765 will inform the using system when all status bytes have been read by setting Main Status Register bit 6 to a zero. The Specify command does not return any status bytes. The Seek and Recalibrate commands themselves do not return any status bytes either but must be followed by a Sense Interrupt Status command which will return status bytes.

It is convenient to write subroutines to send commands to and receive status from the uPD765. These are called CMDPH and RSLTPH respectively in the diagnostic program listing.

COMMAND SEQUENCE DESCRIPTIONS

The following are very brief descriptions of the commands necessary for basic disk controller programming. Details on these and the very sophisticated search and scan commands may be found in the uPD765 data sheet.

Specify

The specify command is used to establish disk system operating parameters that remain constant. It must be executed first after every system reset. The first byte is the command code which is 03. The high nybble of the second byte is the two's complement of the stepping speed to be used on Seek and Recalibrate commands. Thus a value of A would be used for 6MS stepping (167 steps/second). The low nybble specifies how long the head should remain loaded to the disk after a read or write command completion in 16 MS increments. The maximum value (F or 240MS) is normally recommended. The most significant 7 bits of the third byte specifies the head load time allowed for the disk drive in 2 millisecond increments. 40MS should be adequate for any kind of drive. Note that this is also the time allowed for head settling after a seek before attempting to read or write data. The least significant bit of the third byte is normally a zero which specifies the DMA mode of data transfer. The Specify command is executed immediately, does not generate an interrupt, and returns no status bytes.

Recalibrate

The Recalibrate command is used to position the head on the selected disk drive to track 0 without making an assumption about where it is presently. This should be the second command executed after reset but can also be used as part of an error recovery procedure. The first byte of the Recalibrate command contains the command code of 07. The least significant two bits of the second byte specify the drive number in binary of the drive to be recalibrated. The reamining bits should be zero. After sending the Recalibrate command, the disk controller is able to accept additional Recalibrate or Seek commands for the other drives thus allowing simultaneous seek. For simplicity (and less strain on the disk system power supply) however only one seek at a time should be in process.

When the Recalibrate is complete, the Interrupt Request is raised (Hardware Status Register bit 7=0). The program should respond by sending a Sense Interrupt Status command to the disk controller. This is a one byte command with a code of 04. The disk controller will then respond with 2 status bytes. The first byte tells how the Recalibrate command terminated. For normal termination, bits 6 and 7 will be zeroes. Otherwise an error condition occurred and the Status Register 0 table on the uPD765 data sheet should be consulted. Common errors during Recalibrate include failure to reach track 0 after 77 step pulses and the disk drive becoming not ready during its execution. The second status byte is always 0.

Seek

The Seek command is used to position the head to the desired track for reading or writing. The first byte is the command code which is 0F. The least significant bits of the second byte specify the drive number. Bit 2 specifies the head number for two-sided disk drives. The third byte gives the track number to seek to in normal binary code. Execution and completion of the Seek command are the same as with Recalibrate. The second status byte returned by the Sense Interrupt Status command however should be equal to the desired track number.

Read

The Read command is used to read data from the diskette into memory on-board the K-1013 disk controller. If the final destination is on another memory board, a move routine will have to move it there after it is read. Before executing the Read command, the disk head must be on the desired track, the DMA Address Register set to the desired memory address to receive the data, and the DMA direction bit set for write (Hardware Control Register bit 0=1).

Once a Read command is started, the disk controller will continuously read sectors in ascending order until it is stopped by the DMA controller or all of the sectors in the track have been read. Since the DMA controller on the K-1013 does not have a byte count register, the Read command will stop only after the last sector on the track has been read. Fortunately, the Read command itself can specify any number of sectors per track even if it is not the actual number. Thus single sector reads are accomplished by setting the final sector number (EOT) equal to the sector number to read. Although the disk controller will indicate an abnormal command termination, there are no ill effects.

The Read command requires 9 bytes. The first is the command code which also specifies the data density and other information. The normal code of 46 gives double-density, makes no distinction between normal and deleted data address marks, and does not automatically continue a read command from one side to the other on double-sided drives. The second byte specifies the drive number and diskette side in the same format as the Seek command. The third byte must be equal to the track number the head is currently at. The fourth byte is the side number (=0 for single sided drives, 0 or 1 for double-sided drives). The sector number to start reading at is the fifth byte. The sixth byte in conjunction with the ninth byte gives a code for the number of bytes in the sector. For the 256 byte sectors used by CODOS, the sixth byte should be set to 01 and the ninth set to FF. The seventh byte gives the last sector number to read. It should equal the fifth byte to specify a single sector read. The eighth byte gives the gap length between sectors and is normally set to 0E for 26 sectors, 256 bytes/sector, double density.

When the read operation is complete, the controller will request an interrupt and the program should respond by reading status bytes back. (Do <u>not</u> execute a Sense Interrupt Status to get the status.) The first three status bytes (from a total of 7) give an error code. Normal termination in the K-1013 is for the first byte to read 40+4*(head number)+(drive number) which is hex 40 for a single-sided drive 0. The second status byte should read hex 80 and the third should be all zeroes. Anything else indicates some kind of read error and the data sheet should be consulted for the meaning.

Write

In most respects the Write command is identical to the read command. Before executing a write however the DMA direction bit should be set for read (Hardware Control Register bit 0=0). Once a Write command has been started, it will write continuously just like the read command. This is handled in the same way as with the Read command. The normal command code for write is 45 which gives double density and does not continue writing from one side to the other on double-sided disks. The remaining 8 bytes of the command are exactly the same as for the corresponding Read command. Also, execution and the 7 status bytes returned at completion are the same as for Read.

Following the write command a delay of at least 500uS must elapse before attempting to seek. If this delay is not allowed, either the drive will ignore the first step pulse or data on the adjacent track will be damaged by the tunnel erase head which remains energized for 500uS after the write current is turned off.

Format

The Format command is used to erase a disk and write the various address marks necessary to define the sector boundaries. The Format command applies to an entire track although all tracks on a disk need not be formatted the same way. The standard format for CODOS disks is similar to that used on IBM double density systems except that all tracks are double density. This format defines 26 sectors of 256 bytes on each of 77 tracks for a total of 512,512 bytes per diskette side. An alternate format useful for high speed data acqusition (e.g. digital audio) might use 16 sectors of 512 bytes for a total of 630,784 bytes, a 23% increase.

The command code for Format is normally 4D which specifies double density. The second byte specifies the drive and head number as with read and write commands. The third byte is a code for the number of bytes per sector; 01 is used for 256 bytes and 02 is used for 512 bytes. The fourth byte is the number of sectors per track and would normally be 1A hex for 26 decimal sectors per track. The fifth byte gives the gap length between sectors which is normally 36 for double density. The sixth byte is the filler value to which all of the data bytes will be set. CODOS uses EA.

During execution of the Format command, additional formatting data is read from memory via direct memory access. Thus before executing the Format command the DMA direction must be set to read (Hardware Control Register bit 0=0), the DMA address register set, and the formatting data to be described prepared in memory. Each sector to be formatted reads 4 bytes from memory. The first byte must equal the track number that is being formatted. The second byte must equal the side number being formatted (set to 0 for single sided). The third byte gives the sector ID for the sector being formatted. The fourth byte is a code for the sector length and must be equal to the third byte of the command string described above. For formatting 26 sectors then, 4*26 or 104 bytes of formatting data will have to be set up in memory.

Note that the sectors need not be sequentially numbered around the track and each track can be different thus allowing optimized numbering for faster throughput when the disk is later read or written. CODOS format takes advantage of this and uses alternate numbering and staggering from track to track to attain an average throughput of about 20K bytes per second when reading sequential data from disk such as when loading a program. The exact format is described in the CODOS manual.

When the entire track has been formatted (as signalled by the second occurance of the index hole), an interrupt is generated and 7 status bytes may be read. Normal termination is for the first status byte to read 4*(side number)+(drive number) and the second and third status bytes to read zero. As with the Write command, a delay of at least 500uS must elapse before issuing a Seek to the next track so that tunnel erase is completed.

JUMPER OPTIONS

In order to accomodate a variety of system configurations, the K-1013 Disk Controller board has a number of jumper options. In order to simplify, or possibly eliminate, the task of configuring the board for the user's system, the board is shipped with a "standard" jumper configuration that should be suitable for the largest number of users. This standard configuration is listed on page 1 and is also designated with an * in the jumper option tables in this section. There are three classes of jumpers: address selection, system characteristics, and disk characteristics. Each of these will be described in detail in the following paragraphs.

ADDRESS SELECTION JUMPERS

To the using system, the K-1013 Disk Controller looks like 16K bytes of memory. This 16K is actually broken down into two completely independent 8K blocks. One of these blocks, which is called User RAM, is totally free for use as 8K of read/write memory. The other block, which is called System RAM, consists of 7.8K of read/write memory, 248 bytes of read-only memory, and the various I/O port registers associated with the disk controller chip (see the Programming section). This block is normally used to hold the CODOS Disk Operating System although it too can be successfully used as memory if its write protect flip-flop is turned off.

Each block has its own set of address selection jumpers which are small staple-shaped pieces of wire with white insulation and chisel pointed ends. They may be repeatedly plugged into and removed from the jumper sockets without harm. There should be enough jumpers supplied with the board to support any desired reconfiguration. Alternatively, DIP switches may be installed if frequent reconfigurfation is anticipated. All address selection jumpers are installed in the socket labelled U27 (see the assembly diagram on page 64).

ADDRESS RANGE	SYSTEM RAM JUMPERS				USER RAM JUMPERS			
0000 - 1FFF				8-9				4-13
1000 - 2FFF			7-10				3-14	
2000 - 3FFF			7-10	8-9			3-14	4-13
3000 - 4FFF		6-11				2-15		
4000 - 5FFF		6-11		8-9		2-15 *		4-13 *
5000 - 6FFF		6-11	7-10			2-15	3-14	
6000 - 7FFF		6-11	7-10	8-9		2-15	3-14	4-13
7000 - 8FFF	5-12				1-16			
8000 - 9FFF	5-12 *			8-9 *	1-16			4-13
9000 - AFFF	5-12		7-10		1-16		3-14	
A000 - BFFF	5-12		7-10	8-9	1-16		3-14	4-13
B000 - CFFF	5-12	6-11			1-16	2-15		
C000 - DFFF	5-12	6-11		8-9	1-16	2-15		4-13
D000 - EFFF	5-12	6-11	7-10		1-16	2-15	3-14	
E000 - FFFF	5-12	6-11	7-10	8-9	1-16	2-15	3-14	4-13

* Indicates standard jumper installed at the factory.

SYSTEM CHARACTERISTICS JUMPERS

The Vector Fetch Enable jumper is placed between U47 pins 4 and 13 and only has an effect in a KIM-1 system. When installed, the Reset, IRQ, and NMI vectors will be in the KIM monitor ROM as usual. When removed, the vectors refer to the ROM on board the K-1013 (provided that the System RAM is jumpered for E000-FFFF) and thus provides for automatic system loading in KIM-1 systems. SYM-1, AIM-65, and PET systems normally have their own monitor ROM in the vector area of memory and thus Reset will always enter their respective monitors. This jumper is normally installed.

The IRQ Enable jumper is placed between U47 pins 8 and 9. When installed, an interrupt request from the disk controller chip will be wire-ored with other interrupt sources in the system and cause an IRQ sequence in the 6502 to occur if the 6502 Interrupt Disable bit is turned off. The disk controller may be identified as the source of the interrupt by polling the Hardware Status Register on the K-1013 (see Programming section). When this jumper is in place, there is no way to disable interrupts from the disk controller without disabling all IRQ interrupts in the system. The CODOS Disk Operating System does not require interrupts to function and this jumper is normally removed.

The IPL option jumper is placed between U47 pins 6 and 11. Its only function is to set bit 6 in the K-1013 Hardware Status Register to a one if present. It is normally used to inform the Initial Program Load ROM that standard density (as opposed to double density) is being used in the system. This jumper is normally removed which signals double-density operation to the IPL program.

DISK CHARACTERISTICS JUMPERS

The Write Precompensation jumpers are used to control the amount of write precompensation used in double-density operation. These jumpers have no effect on standard density operation but one must be present for the write circuitry to function. 125NS is normally chosen for 8 inch disks and 250NS for 5 inch disks. In some cases with older drives not rated for double density, better results are obtained with 250NS for 8 inch and 500NS for 5 inch.

* U41-1 to U41-16	125NS
U41-2 to U41-15	500NS
U41-3 to U41-14	250NS

The Force 2-Sided jumper is used force the Two Side Sense input to the disk controller chip to a logic one. This in turn sets bit 3 of status register 3 to a one which indicates that a two-sided drive and diskette are present in the system. Its practical function is to allow the use of both sides of single-sided diskettes in two-sided drives (note however that single-sided diskettes are only certified on one side). True 2-sided diskettes have the aperture for the index sensor in a different location so that two-sided drives can sense when two-sided diskettes are being used. This jumper is normally removed.

The Write Clock jumper selects between 8 inch and 5 inch drives. This is not a pluggable jumper; rather it is an etch cut and a soldered-in jumper. The K-1013 is shipped with the write clock set for 8 inch disks. To use 5 inch disks, the line between J1 and J2 must be cut and then a jumper placed between the two J1 points and another placed between the J2 points. Note also that for operation with 5 inch drives that 470pF 5% polystyrene capacitors must be installed at C90, C92, and C94, 1000pF 10% at C86 and .01uF 10% at C96. Also the data separator adjustments should be performed as described in the Adjustment Procedure section.

DISK DRIVE CHARACTERISTICS

The K-1013 Floppy Disk Controller is designed to mate directly with Shugart compatible 8 inch drives. However because of its low cost and unbundled marketing technique, many customers may want to use incompatible drives that they have available. This is indeed possible and MTU has successfully operated a Qume Datatrak 8, a Calcomp model 142 and a Pertec FD400 with the controller. Minifloppies have not been tried but they too should be relatively easy to adapt (be sure to add the 5 capacitors and move the write clock jumper on the K-1013 to slow it down for minifloppy operation).

Fortunately all 8 inch floppy disk drives are very similar in their construction and operation. The major differences are power requirements, signal polarity, and cable connector used. While it may be possible to hack away at the K-1013 to resolve differences in signal polarity, it is strongly recommended that an adaptor board be constructed that makes the drive look Shugart compatible. Typically the board will be quite small containing only two or three connectors and a couple of IC's. In most cases it can simply be plugged into the incompatible drive's signal connector and then a standard cable plugged into the Shugart compatible connector. This technique would then allow mixing of different drive types in the same system.

In the following sections, each of the signals typically found in a floppy disk drive will be discussed giving the range of variations encountered by the author. Then a simple circuit to convert that signal to/from Shugart compatibility will be given where appropriate.

DRIVE SELECT

A Shugart compatible disk drive is intended to be used on a "disk bus" with up to 4 drives attached. When a drive sees its DRIVE SELECT signal go low, it places its status on the bus and receives its commands, if any, from the bus. When its DRIVE SELECT is high, the drive does absolutely nothing. Actually there are 4 DRIVE SELECT lines in the cable, one for each drive. In a Shugart compatible drive there is a jumper that selects which of the 4 lines a particular drive is to respond to and an adaptor board should be built the same way with a jumper socket or dipswitch. If the drive to be used only busses some of the signals or even none of them, high current open-collector bus drivers such as an 88xx should be used on the adaptor. If only a couple of signals are not set up for bussed operation, such as Ready or Head Load, then open collector power gates, such as the 7438, can be used instead. Remember that the 220 ohm pullup to +5 volts at the K-1013 and most disk drives requires a sink current capability of at least 24MA which is beyond the spec limits for standard TTL.

DRIVE MOTOR POWER

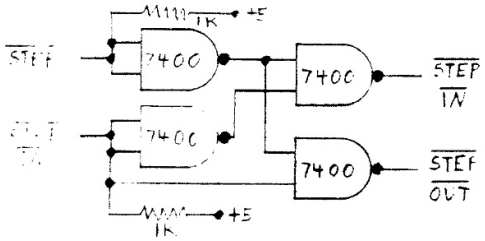
Virtually all 8 inch disk drives run the drive motors as long as AC power is applied. In drives that use a DC motor (such as the Pertec and most 5 inch drives), the motor will have to be wired so that it runs continuously since there is no signal for turning the motor on and off. If a door switch is available, it may be convenient to wire it so that the motor is on only when the door is closed.

STEPPER MOTOR POWER

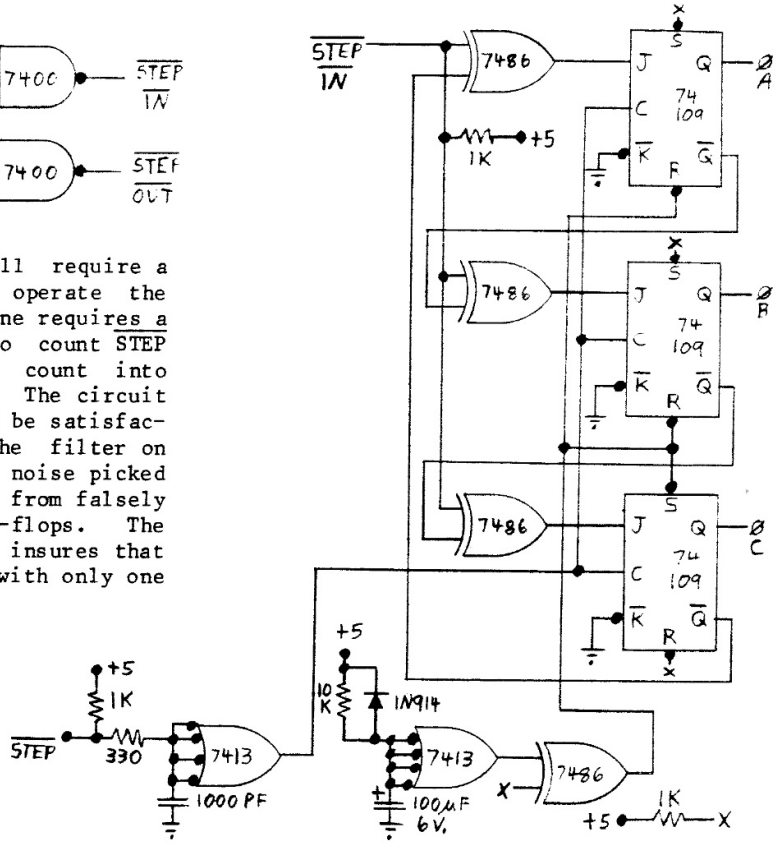
Like drive motor power, the stepper motor generally has to be powered continuously. Some drives internally sense seek pulses and power up the motor for the duration of the seek. This could be simulated on the adaptor board with a retriggerable single-shot set for 50MS or so. It is not acceptable to use DRIVE SELECT to control the stepper power since the 1kHz scanning of the K-1013 will cause a high pitched squeal and possible motor overheating.

STEP, STEP IN

In a Shugart compatible drive, two lines are used to control seeking, The $\overline{\text{STEP}}$ line pulses low to make the drive seek to the adjacent track. The state of the $\overline{\text{STEP IN}}$ line during the $\overline{\text{STEP}}$ pulse determines the direction of seek. When $\overline{\text{STEP IN}}$ is low, the stepping will be in toward the center of the disk. Many drives have signals called $\overline{\text{STEP IN}}$ and $\overline{\text{STEP OUT}}$. Pulsing the $\overline{\text{STEP IN}}$ line will cause seeking toward the center (higher numbered tracks) and pulsing the $\overline{\text{STEP OUT}}$ line will cause seeking toward the periphery. The circuit below can be used to convert the step and direction format to the step-in-step-out format:



Occasionally a drive will require a three phase input to operate the stepper motor. Using one requires a flip-flop arrangement to count $\overline{\text{STEP}}$ pulses and decode the count into 3-phase drive signals. The circuit toward the right should be satisfactory in most cases. The filter on the $\overline{\text{STEP}}$ line prevents noise picked up on the disk cable from falsely triggering the flip-flops. The power-on reset circuit insures that the flip-flops come up with only one motor phase energized.



LOW CURRENT

The current fed to the head during writing must be reduced on tracks 43 and higher. Only the polarity of this signal should cause trouble. The K-1013 provides a low logic level on this line for track 43 and greater and a high level for track 42 and less. Some drives sense the head position internally and therefore do not need this signal. Others simply ignore it.

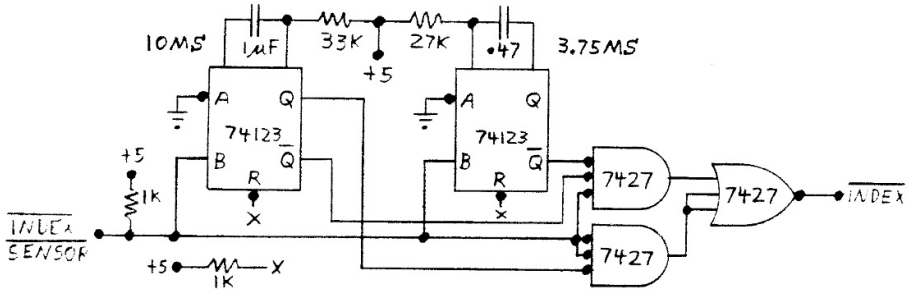
READ DATA, WRITE DATA

There should be no problem with these signals. All drives that have been seen expect low-going pulses on the $\overline{\text{WRITE DATA}}$ line to generate flux reversals and give back low-going pulses on the $\overline{\text{READ DATA}}$ line to mark flux reversals read. If separated data and clock are available, ignore them; the K-1013 expects raw data (clocks and data mixed).

INDEX

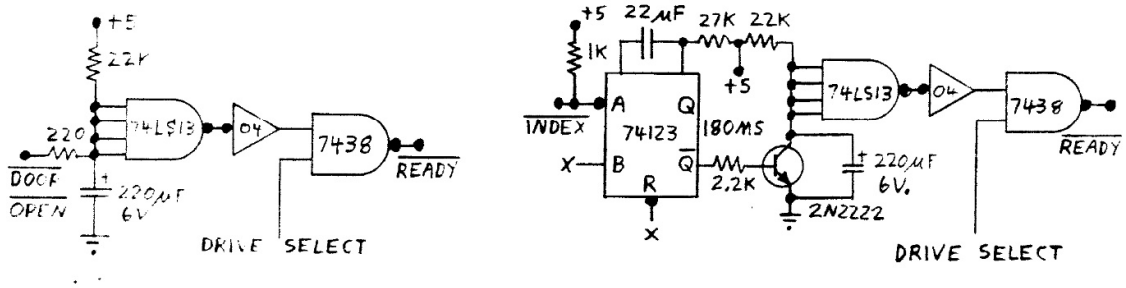
The K-1013 uses a soft sectored disk format. Accordingly it expects to see a single pulse on the INDEX line for every rotation of the disk. Actually the index is only used during the Format command and as a timeout signal to terminate commands when the desired sector cannot be found. If the disk drive has a separated Index and Sector output, the Sector output will normally have the correct signal when a soft-sectored disk is inserted (only one hold punched in the media).

The circuit below can be added to a disk drive (even if it is Shugart compatible) to allow the use of either hard or soft sectored diskettes. This might be useful if one has a large inventory of hard sectored diskettes and no other use for them. Be aware that the circuit only makes the Index pulse compatible, hard sectored formats cannot be read or written by the K-1013.



READY

Most disk drives have a circuit that senses when a diskette is inserted and is up to proper rotational speed. This is normally done with a retriggerable single shot that looks at the index pulses and requires that they have a minimum frequency for a period of time before Ready status is generated. Having a proper Ready signal prevents the K-1013 from becoming "hung up" when trying to read from a drive that is not generating index pulses. Two different Ready circuits are given below. The one on the left may be used with drives that have a door closed switch. The one on the right is more sophisticated and may be used with any drive.

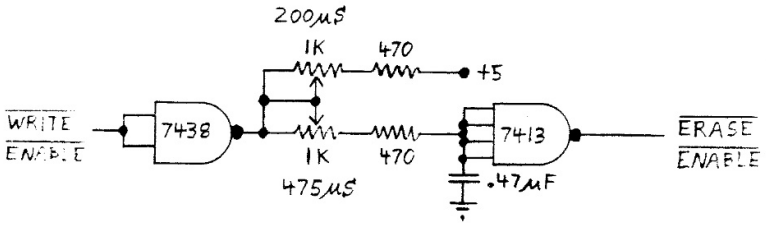


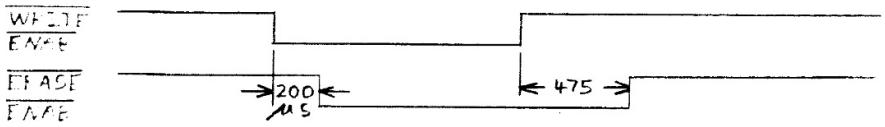
FAULT SENSE AND RESET

Some drives have what is known as a "write fault sensor" which is intended to detect failures in the internal write logic of the drive. If one of the defined fault conditions is detected a flip-flop is set which prevents further use of the drive. Over the years users have found this feature of limited usefullness and Shugart drives do not have the feature. If your drive has a write fault circuit, just ignore it if possible. If absolutely necessary, two of the unused signal lines in the cable can be assigned and the FAULT SENSE and FAULT RESET signals on the K-1013 wired over to them.

ERASE ENABLE

Every floppy disk drive has an "erase" head mounted behind the read/write head. Its function is to trim away the edges of the written track so that slight amounts of misalignment do not affect readback signal amplitude. When writing new data in a sector it is desirable to trim only the data written. Thus because of the physical displacement of the erase head with respect to the read/write head, application and removal of current to the erase head must lag the application and removal of write current. Most disk drives have the necessary delay circuitry built-in but a couple on the market do not. The IBM diskette format standard in conjunction with standard read/write head geometry requires that erase current application be delayed 200uS from write current application and that erase current removal be delayed 475uS from write current removal. The circuit below can be used to perform this function for drives that need an ERASE ENABLE signal. Adjustment is performed by setting R1 for a 475uS trailing edge delay first and then R2 for a 200uS leading edge delay. WRITE ENABLE pulse widths must be at least 5MS and separated by 5MS for this circuit to work properly.





PRINCIPLES OF OPERATION

The K-1013 Disk Controller is basically a 16K memory board with a disk controller IC, direct memory access logic, TTL registers, and 256 bytes of PROM added. In operation it is synchronized to the 6502 bus cycle which is expected to be 1.0MHz to close tolerances. Although there are a lot of parts on the board, its operation is straightforward. In addition to the digital logic, there is a small amount of analog circuitry in the clock generator and data seperator. There should be enough information in this section to allow an experienced technician to understand the board's operation and make repairs and adjustments.

Page 65 shows a block diagram of the K-1013 Disk Controller board. Data are exchanged among the various elements of the board via a central bidirectional data bus. Because of the two-phase bus cycle of the 6502, this internal bus can be devoted to internal operations such as direct memory access and memory refreshing during Phase 1 and then turned over to the 6502 (if the board is addressed) during Phase 2. The memory IC's and the uPD765 disk controller IC are therefore operated at a 2mHz cycle rate which these modern devices handle quite well. The address bus is unidirectional being simply a buffered version of the 6502 address bus and two levels of multiplexors are used to select addresses from one of three sources for the RAM chips themselves.

ADDRESS RECOGNIZER

Because of the large number of different address ranges to be decoded and flexible address jumpering, a fairly large amount of logic located at the right side of page 1 of the schematics is devoted to address recognition. All 16 address lines are first buffered by U1 and U2 which are low power non-inverting octal buffers to produce LOC AB0 through LOC AB15. U29-6 in conjunction with inverters at its input produces KIM DEC ENAB for KIM-1 systems which goes low when addresses in the range of 0000 through 1FFF are on the address bus. U8-8 detects addresses in the range of FF00 - FFFF and generates KIM VECT FETCH when they are seen. Germanium diode D4 in series with its output simulates the open-collector output which is needed. The Vector Fetch Enab jumper between U47-4 and U47-13 allows this signal onto the bus when it is inserted.

The first level of address decoding determines whether the User RAM block or the System RAM block is addressed, if either. This is accomplished with two completely separate circuits. A block recognizer operates by taking the 4 most significant address lines and then adding a 4 bit constant to them with a 4 bit adder. When the sum is either E or F, the recognizer is satisfied. The value of the constant is determined by 4 jumpers which in turn determine the 8K range on 4K boundaries that the recognizer will respond to. U28 in conjunction with U29-8 recognizes addresses intended for the User RAM block while U18 in conjunction with U29-12 recognizes System RAM addresses. Test points are provided for USRRAM ADRD and SYSRAM ADRD to facilitate troubleshooting. U39-8 logically OR's the two signals together to produce a board addressed signal.

U17-8 determines if the on-board PROM might be addressed by looking for an F from the System RAM adder (U18) and LOC AB8 through LOC AB11 to be all ones. U7-8 looks for addresses between xxE8 and xxEF which are candidates for the various I/O registers on the K-1013. U38-12 makes the final decision about whether a System RAM address actually referrs to RAM by requiring that SYSRAM ADRD be true and that the output from U17-8 be false. U38-6 makes the final decision about PROM addresses by requiring that SYSRAM ADRD be true, U17-8 be true, and U7-8 be false. U38-8 makes the final decision about I/O addresses by requiring SYSRAM ADRD be true, U17-8 be true, and U7-8 be true. When I/O addresses are detected by U38-8, U40, which is a 1-of-10 decoder, is enabled which looks at LOC AB0, LOC AB1, and the read/write line to determine exactly which device to activate.

DATA BUS BUFFERS

The data bus is buffered by two octal buffers, U3 and U9. If something on the board is addressed and the 6502 is performing a Read cycle (6502 R/$\overline{W}$ high), U9 is activated during PHASE 2 to drive the 6502 data bus with read data. If something on the board is addressed and the 6502 is performing a Write cycle (6502 R/$\overline{W}$ low), U3 is activated during PHASE 2 to drive the internal K-1013 data bus (LOC DB0 - LOC DB7) with write data. Neither buffer is activated during PHASE 1 in order to reduce noise generation.

HARDWARE CONTROL REGISTER

The Hardware Control Register is in the middle of schematic page 2 and consists of two flip-flops. These are wired up as a D-type register and are clocked by the trailing edge of $\overline{\text{CNTL WRT}}$ which is activated when address 1FE8 in the System RAM is written to. LOC DB0 is written into the bottom flip-flop which controls the direction of DMA data transfers. LOC DB1 is written into the top flip-flop which allows or inhibits writing into the System RAM by the 6502. Reset from the bus is connected to both flip-flops in order to put them into a known state (DMA mode=read, System RAM write disabled) at power-up.

HARDWARE STATUS REGISTER

The Hardware Status Register is at the top left corner of schematic page 3. It consists simply of two tri-state buffers which gate data onto LOC DB6 and LOC DB7 when activated by $\overline{\text{STATUS READ}}$ which in turn responds to read cycles from address 1FE8 in the System RAM. The Interrupt Request line from the uPD765 floppy chip is gated onto LOC DB7. A pluggable jumper determines the logic level gated onto LOC DB6.

DMA ADDRESS COUNTER

The DMA address counter is in the center of schematic page 4. It is a 14 bit counter with the least significant 6 bits being U22 which is an asynchronous clearable counter and the most significant 8 bits being U33 and U43 which are 4 bit synchronous loadable counters. Only 14 bits of counter are necessary because there is only 16K of on-board memory to address.

Normally the counters are set up to count pulses seen on the DMA CYC line. When the most significant bit of the asychronous counter makes a 1-to-0 transition, it is coupled through C77 and U62-6 to clock the synchronous counters which are set up for counting. However when the address decoder generates $\overline{\text{DMA CNT LD}}$, the asynchronous counters are cleared and the synchronous counters are loaded from LOC DB0 - LOC DB7. The delay network formed by R15 and C58 insures that the counters are kept in the Load Enable state long enough after clocking to load properly. U62-6 functions as an OR gate so that the synchronous counters are clocked either by $\overline{\text{DMA CNT LD}}$ or by a carry out from the asychronous counters.

IPL PROM

The IPL ROM is at the left center of schematic page 4. It is activated during read cycles when $\overline{\text{ROM CS1}}$ is true by virtue of its two chip select inputs. Its 8 address inputs are connected directly to LOC AB0 - LOC AB7 for addressing of its 256 bytes. The 8 tri-state outputs are connected directly to the on-board LOC DB bus. A type 6309 (or equivalent) 256 word by 8 bit fusible link PROM is used. Note that locations E8-EF cannot be read from the PROM because the address decoder overlays them with I/O register addresses.

uPD765 FLOPPY DISK CHIP ADDRESSING

The uPD765 Floppy DIsk Controller IC is located at the upper left corner of schematic page 3. It has only two internal registers that must be addressed. LOC AB0 is used to distinguish between the Main Status Register and the Data Register. The uPD765 has separate $\overline{RD}$ and $\overline{WR}$ inputs which are used to strobe data from and into internal registers respectively. The driving signals, $\overline{FDC\ RD}$ and $\overline{FDC\ WRT}$ are generated by the DMA logic which simply OR's DMA access to the uPD765 with outputs from the address decoder for 6502 access. Its bidirectional data bus is simply tied to the on-board LOC DB bus.

TIMING GENERATOR

The timing generator is at the left side of schematic page 2. All timing is generated by counting down an 8mHz clock and then decoding the count in various ways to insure that accurate timing is always generated. The timing diagram on page XX may be consulted for detailed timing relationships.

The timing generator is synchronized to the trailing edge of 6502 PHASE 2 by means of a phase-locked loop which is at the bottom left of the drawing. U57-8 is a simple Schmidt trigger oscillator with a nominal frequency (which can be adjusted with R27) of 8MHz. By connecting R30 to the R-C node, the frequency can be controlled by the application of a DC voltage to its free end. Although the linear control range is only 20% or so, it is ample for locking onto the crystal controlled system clock. The 8MHz output at U57-8 is normally assymetrical (35% high, 65% low) and goes to a number of places including the synchronous 4 bit counter, U21. U31-6 decodes the counter status to produce a low-going signal with duty cycle of 25% at a frequency of 1MHz. This is the comparison signal for the phase comparator. 6502 PHASE 2 is the reference signal and U50 is used as a phase comparator. The output of the phase comparator simply floats for 3/4 of each luS cycle since it is actually a tri-state gate. When it is enabled by U31-6, the output first goes high (since when locked 6502 PHASE 2 would be high), then goes low when 6502 PHASE 2 terminates, and then floats when disabled by U31-6. The ratio of high-to-low time of this signal is averaged by lowpass filter R31, C84, and R29 which is then the control voltage to the 8MHz oscillator.

Normally the trailing edge of 6502 PHASE 2 occurrs midway in the "window" defined by the output of U31-6. Locking action can be understood by considering what would happen if 6502 PHASE 2 terminated later in the window, i.e., slowed down slightly. The output of the phase comparator would then be high for a longer time and low for a shorter time thus raising the averaged control voltage. Since a higher control voltage slows down the oscillator, the window frequency would decrease to match the input. The converse would occur if 6502 PHASE 2 should speed up. R27 can be adjusted to center the trailing edge of 6502 PHASE 2 in the window for accurate timing. This circuit has been found to be highly reliable and is in fact used on virtually all MTU bus interface products to provide a phase locked high frequency clock for timing generation.

The most critical timing signals needed are those that operate the 16K dynamic RAM chips. U42-6 and both halves of U32 are set up as a 3 bit shift register delay line to generate the $\overline{RAS}$, address swap, and $\overline{CAS}$ sequence needed by the memories. A memory cycle is started if MEM CYC ENAB is true when bit 1 of the 4 bit timing counter is low and a positive edge of 8 MHZ CLOCK is seen. U42-6 then goes low generating $\overline{RAS}$ which starts the timing chain for the memory cycle. The next negative edge of 8 MHZ CLOCK flips U32-6 which through the memory address multiplexor, switches from row address to column address. Finally, the next positive edge of 8 MHZ CLOCK after the address is switched generates $\overline{CAS}$ which latches the column address in the RAM chip and activates the RAM I/O circuitry. At the end of the memory cycle, U42-6 is jammed back high by U20-8 and U32-6 and U31-10 follows later in preparation for another memory cycle.

The uPD765 Floppy Disk Controller IC requires a Write Clock signal that is exactly twice the bit rate to the floppy disk and is 250NS wide independent of its frequency. U5-10 acts as a fifth counter bit on the timing generator and in conjunction with U31-3 and U11-6 produce FDC WRT CLK at a frequency of 1.0MHz, .5MHz, or .25MHz depending on the disk type and recording density. With the J1 and J2 jumper set for standard 8" disks, FDC WRT CLK will be .5MHz if MFM MODE is a logic one and will be 1.0MHz if MFM MODE is a logic zero, With the J1 and J2 jumpers set for mini disk operation, both frequencies are divided by two for .25MHz and .5MHz respectively. The pulse width in all cases is 250NS.

Z11 produces a .5uS pulse every 4uS which triggers a refresh cycle for the memory chips. However, if a DMA cycle is in progress the refresh cycle is omitted until the next 4uS period. There is no chance that DMA will lock out refreshing however because the maximum data rate to or from the disk is every 16uS. Even if every DMA cycle preempted a refresh cycle, the refresh rate would only drop from 128 cycles per .5MS to 128 cycles per .62MS.

MEMORY CYCLE CONTROL

The Memory Cycle control circuitry is at the top right of schematic page 2. Its main function is to determine if a memory cycle is needed during PHASE 1 and if so, whether it is a DMA cycle involving the uPD765 chip or a refresh cycle. It also distinguishes between read and write cycles based on the setting of the DMA WRITE and WRITE PROT flip-flops. A DMA control chip, such as an 8257, was not used because it is too slow to control .5uS memory cycles.

U42-10 is used to synchronize DMA requests from the uPD765 to memory cycle oppertunities during PHASE 1. When U42-10 is set, it sends an acknowledge back to the uPD765 ($\overline{\text{FDC DACK}}$) which then drops its request. The request is dropped soon enough so that two DMA cycles are not taken. U52-11 logically AND's the DMA flip-flop with CNTC (which is approximately the same as PHASE 1) and generates DMA CYC and $\overline{\text{DMA CYC}}$ through inverter U53-12. If the DMA mode is read, U52-3 will generate a write pulse coincident with DMA CYC for the uPD765 which is OR'ed with write pulses from the 6502 in U62-3. If the DMA mode is write, U52-6 will generate a read pulse which is OR'ed with reade pulses from the 6502 in U62-11. The delay network consisting of R52 and C98 prevents possible overlap of the uPD765 driving the LOC DB bus before the RAM chips have stopped driving it.

Considerable logic is required to generate $\overline{\text{RAM WE}}$ only when needed. U52-8 requests the generation of $\overline{\text{RAM WE}}$ during all DMA cycles when the DMA mode is write. U30-6 requests $\overline{\text{RAM WE}}$ when the System RAM is addressed by the 6502, a 6502 write cycle is in progress, and the Write Protect is off. CNTC is factored in to include only 6502 cycles during PHASE 2. U20-6 requests $\overline{\text{RAM WE}}$ when the User RAM is addressed by the 6502 and a 6502 write cycle is in progress. All of these requests are OR'ed together in U20-12. The result is AND'ed with RAS to produce a properly timed $\overline{\text{RAM WE}}$ pulse for the RAM chips.

The circuitry at the top right of schematic page 1 also controls memory cycles. U30-8 OR's cycle requests from 4 different sources to produce MEM CYC ENAB which then triggers the timing generator to execute a memory cycle. Two of the sources, $\overline{\text{DMA CYC}}$ and $\overline{\text{REF CYC}}$ have already been covered. The third source is User RAM Addressed and the fourth is System RAM Addressed AND'ed with PROM not addressed.

REFRESH ADDRESS COUNTER

The refresh address counter is at the bottom left corner of page 4. It is simply an 8 bit asynchronous counter that counts up on the trailing edge of $\overline{\text{REF CYC}}$ pulses. The eighth bit is not needed to refresh 16K RAM chips.

MEMORY ADDRESS MULTIPLEXOR

The Memory Address Multiplexor is at the bottom right of schematic page 4. It is actually a two-level multiplexor and simultaneously performs address selection from 3 different sources and row/column multiplexing for the RAM chips. U13 and U23 form the first multiplex level which selects between the refresh address and the low 7 bits of the DMA address. Note that the high 7 bits of the DMA address will actually be the high 7 bits of the refresh address but these address bits are ignored during refresh cycles.

The output of the first level multiplexor along with the upper 7 DMA address bits and all 14 of the 6502 address bits enters a 4 input 7 bit multiplexor made from U14, U24, U34, and U44. One of the select inputs to this multiplexor selects between 6502 address and DMA/refresh address under the control of CNTC. Thus the 6502 address is selected during PHASE 2 and the DMA/refresh address is selected during PHASE 1. (Actually there is some timing skew because CNTC is not the exact inverse of PHASE 2.) The other select input is connected to the MUX ROW/$\overline{\text{COL}}$ output of the timing generator which selects between the lower 7 address bits (row address) and upper 7 address bits (column address) at the proper point in the memory cycle. Since only 8 type 4116 RAM's are being driven, the bare outputs of the 74LS153's in the multiplexor are sufficient to drive the address inputs.

MEMORY ARRAY

The actual memory matrix is at the top of schematic page 4. All lines to the memory chips exceot data input and data output are simply wired in parallel. One .luF capacitor per chip on both the +12 and -5 power supply busses and a gridded ground network on the board minimize noise generation. Since "early write" timing is used on the RAM's, their data input and data output lines can simply be connected to the on-board bidirectional LOC DB bus.

FLOPPY DISK INTERFACE

Unfortunately the inputs and outputs of the uPD765 Floppy Disk Controller IC cannot be simply connected to a floppy disk drive. Instead, most signals must be buffered and some must be multiplexed because of pin count limitations of the uPD765. In a system with several disk drives, all are parallel bussed and only one at a time is selected by the K-1013 for operation. U61 is a decoder that accepts the two Unit Select outputs of the uPD765 and pulls one of the DRV SEL lines low in response. The drive responds by sending status back to the K-1013 and accepting commands, if any, from the K-1013. Except during read and write operations, all 4 drives (even if they are not installed) and being scanned at a 1KHz rate for status changes.

Four of the uPD765 pins are multiplexed to give the effect of 8 in two groups of 4. The signals that are multiplexed have been chosen so that 4 of them are significant only during read/write operations and the other 4 are significant only during seeking and idle periods. A signal from the uPD765 called RW/SEEK determines which group is active. The signals within a group are further divided into 2 input signals (drive to uPD765) and 2 output signals. Quad tri-state inverters U50 and U59 are appropriately wired to do the necessary multiplexing and demultiplexing with high output drive capability. The 4 input signals that are multiplexed are $\overline{\text{WRITE PROT}}$, $\overline{\text{TRACK 00}}$, $\overline{\text{2-SIDE SENSE}}$, and $\overline{\text{FAULT SENSE}}$. The latter signal is not normally present on Shugart compatible drives so it simply goes to a pad on the K-1013. The 4 output signals that are multiplexed are $\overline{\text{LOW CURRENT}}$, $\overline{\text{STEP}}$, $\overline{\text{STEP IN}}$, and $\overline{\text{FAULT RESET}}$. Like $\overline{\text{FAULT SENSE}}$, $\overline{\text{FAULT RESET}}$ simply goes to a pad. The $\overline{\text{READY}}$ status from the disk drive is not multiplexed and is simply inverted by U51-8 before going to the uPD765. INDEX is likewise inverted by U51-10 and sent directly to the uPD765.

Three outputs from the uPD765 are buffered and sent to the disk drive. These are HDL (HeaD Load), HD (HeaD select for double-sided drives), and WE (Write Enable). The interrupt request output goes to the Hardware Status gate (U59), and also through open-collector inverter U60 to the 6502 IRQ bus line if the IRQ ENAB jumper is installed. The 8MHz clock input to the uPD765 is buffered and given an appropriate duty cycle by U51-6. Pullup resistor R52 gives the somewhat higher logic 1 level required by the uPD765.

WRITE PRECOMPENSATION

During readback of double-density data from diskette, the recorded pulses experience a predictable shift away from their ideal positions in time which make data decoding errors more likely. This tendency may be counteracted by shifting the pulses in the opposite direction when written which is called precompensation. The uPD765 internally computes whether a pulse should be written early, on-time, or late and supplies this information on the PS0 and PS1 lines to the write logic.

U48 is connected as a 3 bit shift register which shifts the FDC WRT DATA output of the uPD765 at a rate determined by the PRECOMP jumper currently in use. Thus the A, B, and C outputs of the register represent early, on-time, and late pulse timing respectively. Multiplexor U58 selects one of these pulses under control of PS0 and PS1 and sends the result through U60-12 for buffering to the disk drive.

DATA SEPARATOR

More than any other portion of a disk controller the data separator determines the overall reliability of the disk system. Not supprisingly, it is also the most difficult circuit to design. The data separator used on the K-1013 is a true analog phase locked loop with none of the time quantization error found in digital phase locked loops.

Actually the uPD765 does the separation of data pulses from clock pulses internally but it must have a clock input (called FDC DATA WINDOW) synchronized to the pulse stream to function. This clock input is a square wave phased to the pulse stream such that pulses occur near the center of half-cycles of the clock. Internally the uPD765 simply registers whether a pulse occurred during a half-cycle or not and the data is separated based on the pattern of pulse-no pulse seen. If a pulse occurred for every half-cycle of the clock, then any of a variety of integrated circuit phase locked loops could be used. Unfortunately they all fail when presented with inputs having "missing" pulses.

Since the goal is to adjust the frequency of the clock so that pulses occurr midway in its half-cycles, a phase comparator based on the difference between the actual position of the pulse and its ideal position can be constructed. If a pulse is missing during a half-cycle, the frequency of the clock should be left alone. In order to distinguish ahead of time the difference between a very late pulse and a missing one, the input pulses need to be delayed by 1/2 of the window width. This is accomplished by U64-13 which produces a 1.0uS output for single density tracking and U64-5 which produces a .5uS output for double density frequency tracking. Both pulse widths may be doubled for mini floppy applications by adding C90 and C92.

The timing diagram on page 40 tells better than words how the phase comparator works. Basically, flip-flop U55-7 is turned on by the data pulse if it is early and turned off at the center point of the window. Thus its on-time is proportional to how early the pulse is. If the pulse is late instead, U55-10 instead turns on at the center point and turns off when the pulse actually appears giving a pulse width proportional to how late the pulse was. If there was no pulse at all (single-shot not triggered), then neither flip-flop is turned on and no frequency adjustment is made.

C97 acts as an integrator (type of low-pass filter) for the early and late pulses from the U55 flip-flops. R26 and C87 "shapes" the response of the filter to prevent excessive overshoot when the loop is locking up. When not disturbed (no pulses within the window), it will retain its charge and thus specify a constant frequency from the voltage controlled oscillator. R51 and R25 do however very gradually pull C97 to +2.5 volts and the oscillator to a "center frequency" value when pulses are absent for long periods. If U55-7 momentarily goes low because a pulse arrived early, D6 becomes reverse biased which allows some of the charge on C97 to go to ground through D7 and R23. This causes a slight reduction in voltage on C97 which will speed up the oscillator to counteract the early pulse. Conversely, if U55-10 goes high because of a late pulse, some charge is added to C97 through D8 and R24 which slows down the oscillator.

Q1, R50, Q2, and R48 form a simple unity gain buffer amplifier to match the high impedance level at the base of Q1 to the low impedance level at the control voltage input to the oscillator. The complementary NPN and PNP transistors reduce the input-to-output offset voltage of the amplifier to about 100MV. U56-8, R49, R46, and C95 form a Schmidt trigger oscillator which is made voltage-controlled by the addition of R47. The nominal frequency of the oscillator is 2MHz which can be adjusted with R49. It is important that the other half of U56 only be used in circuitry that is synchronous to the oscillator.

U54 is used as a 1 bit counter followed by a two bit counter. The single/double density signal from the uPD765 selects either the 1MHz output of the first counter for single density or the 2MHz output of the oscillator for double density. This is accomplished with AND-OR-INVERT gate U63-6 which is wired up as a simple 2 input multiplexor. The output of the 2 bit counter is the actual window signal to the disk controller while the B output is used to establish the center point of the window for use by the phase comparator. U63-8 selects either the 1.0uS single-shot output for single-density or the .5uS single shot for double-density. C66, R16, and R17 convert the wide data pulses from the disk drive into very narrow pulses for the disk controller and phase comparator. Since the entire pulse must be within the data window, narrow pulses are preferred for maximum tolerance of pulse position deviation without error.

POWER SUPPLY

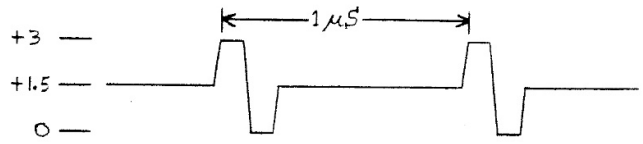
The on-board power supply is at the bottom right corner of schematic page 2. Positive 5 volts for the board logic is derived from the unregulated +8 input by VR2 which is a 1 amp 5 volt IC regulator. C11 prevents oscillation and numerous .05uF bypass capacitors maintain regulation during transient current drains. Positive 12 volts for the memory IC's is provided by VR1. C7 suppresses oscillation and provides the addition filtering needed by MTU K-series power supplies on their 16 volt unregulated outputs. C10 absorbs the large current transients typical of dynamic memories. Negative 5 volts for the memory chips is provided by a charge pump circuit consisting of C9, D2, D3, and C8. The circuit is driven from a 12 volt amplitude 1MHz square wave provided by open-collector inverter U60-8 and R6. Shunt regulator D1 limits the negative voltage to -5 volts and in so doing limits the swing at U60-4 to about 6 volts. D1 also prevents the -5 bus from becoming substantially positive during component failure and thus prevents possible damage to the memory chips.

ADJUSTMENT PROCEDURE

Four adjustments are provided on the K-1013 to optimize performance. All are normally made at the factory and the trim pots are sealed with a black adhesive dot. If after troubleshooting or replacing components readjustment is required, please read the appropriate sections. An oscilloscope is required to make the adjustments accurately. A 10MHz DC coupled triggered sweep scope is adequate.

8 MHZ CLOCK (BUS SYNC)

All timing is derived from the 1.0MHz system PHASE 2 clock via a times 8 phase locked frequency multiplier. With the K-1013 in the system receiving PHASE 2 and powered up, connect the scope to TP13 at the top left corner of the board. One should see a doublet pulse as pictured below with a repetition rate of 1MHz. If an unsynchronized mess is seen or the positive and negative portions of the pulse are not of equal width, rotate R27 until the proper waveform is seen. Double-check the frequency to be certain that the loop has not locked on a subharmonic.



DATA SEPARATOR SINGLE-SHOTS

To adjust the single-shots, first disconnect the disk drives and jumper U5-10 to pin 46 on the disk cable connector (J1). This is a .25MHz square wave that simulates data pulses from the disk. Next connect the scope to TP16 which should display positive-going pulses approximately 1uS wide and exactly 4uS apart (you may calibrate the scope sweep to the 4uS if needed for maximum accuracy in the pulse width adjustment). Adjust R42 (which has a fairly limited range) until the pulse width is exactly 1.0uS. If capacitors have been added for minifloppy operation, adjust for exactly 2.0uS. To adjust the double-density single-shot, connect the scope to TP12 and adjust R45 for exactly .5uS (1.0uS for minifloppy).

DATA SEPARATOR VCO ADJUSTMENT

The center frequency of the VCO in the data separator needs to be adjusted to 2MHz for best data recovery reliability. When making this adjustment, there should be no read data pulses coming into the K-1013. If necessary the disk drives should be disconnected. With power on and the scope connected to U54-12, adjust R49 until a 1MHz square wave is seen. The VCO output itself should not be probed during adjustment since the added load capacitance will shift its frequency slightly. If C94 has been added for minifloppy operation, adjust R49 for a .5MHz square wave. After adjustment, double check the DC error voltage at TP17 which should be 2.5 volts. If it is not 2.5 volts or it is varying, verify that pulses are not being received on the READ DATA line.

TROUBLESHOOTING

Before being shipped, all K-1013 are actually plugged into a system and the disk diagnostic in the back of this manual run. CODOS is also loaded and executed. Finally the on-board memory is checked with a memory test program for 24 hours with no errors allowed. Thus when properly connected to the system and to a functioning disk drive, there should be few if any problems.

MEMORY FAILURES

If the board is failing to respond to its address, double check the jumper settings. Also make sure that all of the bus signals, particularly the 16 address lines, needed are actually connected. Try looking at every 4K block in the system to determine if the K-1013 is responding to an incorrect address. If such is the case and the jumpers have been moved, then one of the adder chips U18 or U28 is probably bad. If it appears that the K-1013 has not synchronized to the bus cycle, verify that PHASE 2 is indeed 1.0MHz (±1%). If it is in tolerance, make the 8mHz clock adjustment (see Adjustments section).

If the memory test program fails and the addressing jumpers have been changed, make sure it is testing the right addresses. If the error is consistantly at one bit, then the corresponding memory chip should be replaced. Use a 4116 type 4K RAM with a 200NS speed or less. If the errors affect several bits in a seemingly random pattern but read and write from the monitor seems to work, an address line in the RAM array may have shorted to another line through a bent over pin or embedded metal fragment.

DISK FAILURES

The most common problems are related to disk drive connection. If the Level 1 disk test does absolutely nothing when executed, then disk drive 0 is probably not ready. Most drives require a diskette to be inserted and the door closed before becoming ready. Some may have internal jumpers to select between hard and soft sectoring and their ready circuitry might not work unless configured for soft sectoring.

If the error code indicates that the Recalibrate command failed, then either the drive is not recognizing step pulses or the track 0 sensor is not working. If the drive gave an initial buzz and the head moved away from the center of the diskette, then the problem is the track 0 sensing. If the random seek test failed, the disk drive may not be able to keep up with the step frequency used or its head settling time may be longer than 20 milliseconds. Try a slower seek speed (see program listing). If a step-in-step-out to step-and-direction translator was constructed, step-in may not be working.

If the Format test (Level 2) fails or hangs up, the likely cause is no index pulses reaching the K-1013. Make sure the diskette is not inserted backwards! If no sectors can be read on the read test, look at the error code and uPD765 status bytes. If the status bytes read 40 04 00 or 40 01 01 then it is likely that the Format test did not write anything. Verify that Write Enable is reaching the disk drive and that a write permit tab is put on the diskette (if the diskette has the necessary notch and the drive has the write protect feature). If the read error rate seems excessive, make sure the diskette and drive is rated for double-density. If the problem persists, check the single-shot and VCO adjustments in the data separator.

If all parts of the disk test program operate satisfactorily but the CODOS distribution diskette cannot be read, the drive is probably out of alignment. See the service manual supplied with the drive for alignment instructions.

PROCESSOR INTERFACE

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

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

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

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

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

The bytes of data which are sent to the μPD765 to form the Command Phase, and are read out of the μPD765 in the Result Phase, must occur in the order shown in the Command Table. That is, the Command Code must be sent first and the other bytes sent in the prescribed sequence. No foreshortening of the Command or Result Phases are allowed. After the last byte of data in the Command Phase is sent to the μPD765, the Execution Phase automatically starts. In a similar fashion, when the last byte of data is read out in the Result Phase, the command is automatically ended and the μPD765 is ready for a new command.

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

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

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

② A_0 should equal binary 1 for all operations.

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

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

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

COMMAND SYMBOL DESCRIPTION

SYMBOL	NAME	DESCRIPTION
A_0	Address Line 0	A_0 controls selection of Main Status Register (A_0 = 0) or Data Register (A_0 = 1).
C	Cylinder Number	C stands for the current selected Cylinder (track) number 0 through 76 of the medium.
D	Data	D stands for the data pattern which is going to be written into a Sector.
D_7-D_0	Data Bus	8-bit Data Bus, where D_7 stands for a most significant bit, and D_0 stands for a least significant bit.
DTL	Data Length	When N is defined as 00, DTL stands for the data length which users are going to read out or write into the Sector.
EOT	End of Track	EOT stands for the final Sector number on a Cylinder.
GPL	Gap Length	GPL stands for the length of Gap 3 (spacing between Sectors excluding VCO Sync. Field).
H	Head Address	H stands for head number 0 or 1, as specified in ID field.
HD	Head	HD stands for a selected head number 0 or 1. (H = HD in all command words.)
HLT	Head Load Time	HLT stands for the head load time in the FDD (2 to 256 ms in 2 ms increments).
HUT	Head Unload Time	HUT stands for the head unload time after a read or write operation has occurred (0 to 240 ms in 16 ms increments).
MF	FM or MFM Mode	If MF is low, FM mode is selected, and if it is high, MFM mode is selected.
MT	Multi-Track	If MT is high, a multi-track operation is to be performed. (A cylinder under both HD0 and HD1 will be read or written.)

COMMAND SYMBOL DESCRIPTION (CONT.)

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

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

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

The DIO and RQM bits in the Status Register indicate when Data is ready and in which direction data will be transferred on the Data Bus.

STATUS REGISTER IDENTIFICATION

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

STATUS REGISTER IDENTIFICATION (CONT.)

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

READ DATA

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

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

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

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

Table 1. Transfer Capacity

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

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

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

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

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

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

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

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

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

FUNCTIONAL DESCRIPTION OF COMMANDS (CONT.)

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

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

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

Table 2: ID Information When Processor Terminates Command

WRITE DATA

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

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

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

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

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

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

WRITE DELETED DATA

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

READ DELETED DATA

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

READ A TRACK

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

This command terminates when EOT number of sectors have been read ($EOT_{max} = FF_{hex} = 255_{dec}$). If the FDC does not find an ID Address Mark on the diskette after it encounters the INDEX HOLE for the second time, then it sets the MA (missing address mask) flag in Status Register 1 to a 1 (high), and terminates the command. (Status Register 0 has bits 7 and 6 set to 0 and 1 respectively).

READ ID

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

FORMAT A TRACK

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

After formatting each sector, the processor must send new values for C, H, R, and N to the μPD765 for each sector on the track. The contents of the R register is incremented by one after each sector is formatted, thus, the R register contains a value of R + 1 when it is read during the Result Phase. This incrementing and formatting continues for the whole cylinder until the FDC encounters the INDEX HOLE for the second time, whereupon it terminates the command.

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

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

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

Table 3

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

② Suggested values of GPL in format command.

FUNCTIONAL DESCRIPTION OF COMMANDS (CONT.)

SCAN COMMANDS

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

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

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

Table 4

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

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

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

SEEK

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

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

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

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

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

RECALIBRATE

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

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

SENSE INTERRUPT STATUS

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

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

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

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

Table 5

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

SPECIFY

The Specify Command sets the initial values for each of the three internal timers. The HUT (Head Unload Time) defines the time from the end of the Execution Phase of one of the Read/Write Commands to the head unload state. This timer is programmable from 0 to 240 ms in increments of 16 ms (00 = 0 ms, 01 = 16 ms, 02 = 32 ms, etc.). The SRT (Step Rate Time) defines the time interval between adjacent step pulses. This timer is programmable from 1 to 16 ms in increments of 1 ms (F = 1 ms, E = 2 ms, D = 3 ms, etc.). The HLT (Head Load Time) defines the time between when the Head Load signal goes high and when the Read/Write operation starts. This timer is programmable from 2 to 256 ms in increments of 2 ms (00 = 2 ms, 01 = 4 ms, 02 = 6 ms, etc.).

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

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

SENSE DRIVE STATUS

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

INVALID

If an invalid command is sent to the FDC (a command not defined above), then the FDC will terminate the command after bits 7 and 6 of Status Register 0 are set to 1 and 0 respectively. **A Sense Interrupt Status Command must be sent after a Seek or Recalibrate Interrupt, otherwise the FDC will consider the next command to be an Invalid Command.**

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

SPECIFICATIONS

PHYSICAL: 7 1/2" high by 11" wide exclusive of edge fingers; fits the K-1005 series of card files.

POWER REQUIREMENT: +8 volts unregulated 600MA, +16 volts unregulated 125MA.

BUFFERING: Maximum of 1 LS TTL load on address and data bus lines.

LSI CHIPS USED: uPD765 disk controller, 4116 dynamic RAM 200NS access time.

SYSTEM CLOCK FREQUENCY: Phase 2 clock frequency must be 1.0MHz crystal controlled as provided by all KIM, SYM, AIM, and PET computers.

DISK DRIVE INTERFACE: Shugart 8 inch SA800 compatible drives may be used directly. Others accomodated by jumper and cable change.

ON BOARD MEMORY: 2 independent 8K blocks of RAM, 256 bytes bipolar PROM.

DATA TRANSFER METHOD: Direct access into on-board memory.

WAIT STATES: None under any condition. DMA transfers and memory refresh performed during phase 1 of the system clock. (6502 accesses only during phase 2.

ADDRESSING: Each 8K block of memory may be independently addressed on any 4K boundary. PROM and I/O registers are part of one 8K block.

DATA SEPARATOR: True analog phase-locked loop, no quantizing error.

ADJUSTMENTS: 4, Master clock sync, data separator oscillator, standard density pulse width, double density pulse width.

PIN CONNECTIONS

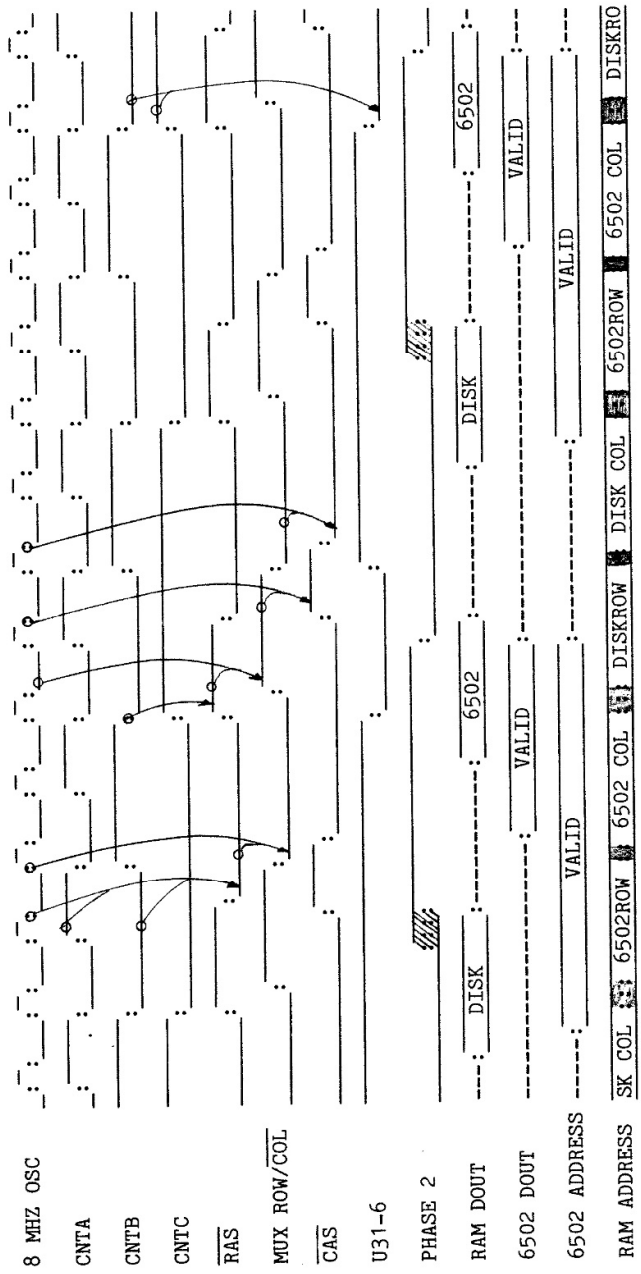
KIM BUS CONNECTOR		SIGNAL
1		N. C.
2		N. C.
3		N. C.
4		IRQ
5		N. C.
6		N. C.
7		RESET
8		DATA BUS BIT 7
9		DATA BUS BIT 6
10		DATA BUS BIT 5
11		DATA BUS BIT 4
12		DATA BUS BIT 3
13		DATA BUS BIT 2
14		DATA BUS BIT 1
15		DATA BUS BIT 0
16		N. C.
17		N. C.
18	*	+8 VOLTS UNREG
19	*	VECTOR FETCH
20	*	DECODE ENABLE
21		N. C.
22		GROUND

KIM BUS CONNECTOR		SIGNAL
A		ADDRESS BUS 0
B		ADDRESS BUS 1
C		ADDRESS BUS 2
D		ADDRESS BUS 3
E		ADDRESS BUS 4
F		ADDRESS BUS 5
H		ADDRESS BUS 6
J		ADDRESS BUS 7
K		ADDRESS BUS 8
L		ADDRESS BUS 9
M		ADDRESS BUS 10
N		ADDRESS BUS 11
P		ADDRESS BUS 12
R		ADDRESS BUS 13
S		ADDRESS BUS 14
T		ADDRESS BUS 15
U		PHASE 2
V		READ/WRITE
W		N. C.
X	*	+16 VOLTS UNREG
Y		N. C.
Z		N. C.

* Do not connect to processor board.

DISK DRIVE CONNECTOR	SIGNAL
1	GROUND
2	LOW WRITE CURRENT
3	GROUND
4	N. C.
5	GROUND
6	N. C.
7	GROUND
8	N. C.
9	GROUND
10	2-SIDED SENSE (2-side only)
11	GROUND
12	N. C.
13	GROUND
14	HEAD SELECT (2-side only)
15	GROUND
16	N. C.
17	GROUND
18	HEAD LOAD
19	GROUND
20	INDEX
21	GROUND
22	READY
23	GROUND
24	N. C.
25	GROUND

DISK DRIVE CONNECTOR	SIGNAL
26	DRIVE SELECT 0
27	GROUND
28	DRIVE SELECT 1
29	GROUND
30	DRIVE SELECT 2
31	GROUND
32	DRIVE SELECT 3
33	GROUND
34	STEP DIRECTION=IN
35	GROUND
36	STEP
37	GROUND
38	WRITE DATA
39	GROUND
40	WRITE ENABLE
41	GROUND
42	TRACK 0
43	GROUND
44	WRITE PROTECT
45	GROUND
46	RAW READ DATA
47	GROUND
48	N. C.
49	GROUND
50	N. C.



MEMORY CYCLE TIMING DIAGRAM

DATA SEPARATOR TIMING DIAGRAM

```
                        .PAGE  'EQUATES AND DATA STORAGE'
                ;       MEMORY TEST AND EXERCISE PROGRAM FOR THE K-1013 DISK CONTROLLER
                ;           THE TEST IS A MEMORY FUNCTION TEST.  RANDOM BITS ARE STORED
                ;       IN A SCRAMBLED ORDER WHICH IS ALSO RANDOMLY DETERMINED.  AFTER
                ;       EVERY MEMORY LOCATION IS FILLED, THE SAME DATA AND SEQUENCE IS
                ;       REGENERATED AND MEMORY CONTENTS ARE CHECKED AGAINST IT.  THEN A
                ;       NEW SEQUENCE IS TRIED.  THIS IS ITERATED 16 TIMES WITH A 16
                ;       SECOND PAUSE BETWEEN THE WRITE AND VERIFY PHASE OF THE 16TH
                ;       ITERATION INSERTED TO VERIFY THE FUNCTIONALITY OF DYNAMIC RAM
                ;       REFRESH.  FOLLOWING THIS ANOTHER GROUP OF 16 ITERATIONS IS DONE
                ;           THIS PROGRAM IS SPECIFICALLY WRITTEN TO TEST THE 15.75K OF
                ;       MEMORY ON-BOARD THE K-1013 DISK CONTROLLER.
                ;       AN ADDITIONAL PAGE (256 BYTES) OF MEMORY IS NEEDED TO
                ;       ACCOMODATE THE TEST ALGORITHM USED.  THIS IS NORMALLY THE PAGE
                ;       FROM 0300-03FF.
                ;       IF THE K-1013 USER RAM DOES NOT START AT 4000, SET THE STARTING
                ;       PAGE NUMBER INTO LOCATION 0200.
                ;       IF THE K-1013 SYSTEM RAM DOES NOT START AT 8000, SET THE
                ;       STARTING PAGE NUMBER INTO LOCATION 0201.

                ;       BASE PAGE DATA STORAGE

0000                    .=      0

0000            SCMEMA: .=.+    2               ; SCRAMBLED MEMORY ADDRESS AND ERROR ADDRES
0002            ERRDTA: .=.+    1               ; INCORRECT DATA READ BACK
0003            OKDTA:  .=.+    1               ; CORRECT DATA WRITTEN
0004            ITCNT:  .=.+    1               ; ITERATION COUNT
0005            RANDNO: .=.+    2               ; RANDOM NUMBER REGISTER
0007            SEED:   .=.+    2               ; SAVES SEED FOR VERIFY
0009            ADDRCT: .=.+    2               ; DOUBLE BYTE ADDRESS COUNTER

000B                    .=      X'200           ; START PROGRAM CODE AT 200

0200 40         USRRAM: .BYTE   X'40            ; PAGE ADDRESS OF USER RAM
0201 80         SYSRAM: .BYTE   X'80            ; PAGE ADDRESS OF SYSTEM RAM
0202 03         FILRAM: .BYTE   X'03            ; PAGE ADDRESS OF FILLER RAM

0203 D8         MTEST:  CLD                     ; INSURE BINARY ARITHMETIC
0204 A901               LDA     #X'01           ; INITIALIZE RANDOM NUMBER REGISTER
0206 8505               STA     RANDNO
0208 A900               LDA     #0              ; ESTABLISH ADDRESS OF HARDWARE CONTROL
020A 8500               STA     SCMEMA          ; REGISTER AND TURN MEMORY PROTECT OFF
020C AD0102             LDA     SYSRAM
020F 18                 CLC
0210 691F               ADC     #X'1F
0212 8501               STA     SCMEMA+1
0214 A0E8               LDY     #X'E8
0216 A900               LDA     #0
0218 9100               STA     (SCMEMA),Y

                ;       TEST: 16 PASSES WITH RANDOM DATA, PAUSE IN 16TH PASS

021A A90F       MAIN10: LDA     #15             ; SET 16 ITERATION COUNT
021C 8504               STA     ITCNT
```



```
021E 20C202     MAIN11: JSR     RAND            ; NEW PASS, GET A RANDOM
0221 A505               LDA     RANDNO          ; NUMBER IN RANDNO AND SAVE
0223 8507               STA     SEED            ; AS SEED FOR VERIFY
0225 A506               LDA     RANDNO+1
0227 8508               STA     SEED+1
0229 205E02             JSR     RNDGEN          ; GENERATE A RANDOM DATA PATTERN IN 16K
022C A504               LDA     ITCNT           ; TEST IF LAST PASS
022E D011               BNE     MAIN15          ; SKIP OVER WAIT IF NOT
0230 A200               LDX     #0              ; WAIT FOR ABOUT 15 SECONDS IN A TIGHT LOOP
0232 A000       MAIN12: LDY     #0              ; TO TEST REFRESH CIRCUITRY SINCE TESTING
0234 A921       MAIN13: LDA     #33             ; IS NORMALLY FAST ENOUGH TO KEEP THE
0236 18         MAIN14: CLC                     ; MEMORY REFRESHED.
0237 69FF               ADC     #-1
0239 D0FB               BNE     MAIN14
023B 88                 DEY
023C D0F6               BNE     MAIN13
023E CA                 DEX
023F D0F1               BNE     MAIN12
0241 A507       MAIN15: LDA     SEED            ; RESTORE RANDOM SEED FOR VERIFY PHASE
0243 8505               STA     RANDNO
0245 A508               LDA     SEED+1
0247 8506               STA     RANDNO+1
0249 207B02             JSR     RNDVER          ; VERIFY
024C D007               BNE     RNERLG          ; GO TO ERROR LOG IF ERROR
024E C604               DEC     ITCNT           ; DECREMENT AND CHECK ITERATION COUNT
0250 10CC               BPL     MAIN11          ; LOOP UNTIL 16 ITERATIONS DONE
0252 4C1A02             JMP     MAIN10          ; REPEAT THE ENTIRE TEST WITH DIFFERENT
                                                ; DATA

0255 8502       RNERLG: STA     ERRDTA          ; STORE ERROR BITS
0257 A505               LDA     RANDNO          ; STORE OK BITS
0259 8503               STA     OKDTA
025B 00                 BRK                     ; BRK TO HALT
025C 0000               .BYTE   0,0             ; ROOM TO INSERT A JUMP INSTEAD

                ;       RANDOM PATTERN STORED IN SCRAMBLED ORDER GENERATE ROUTINE

025E A900       RNDGEN: LDA     #0              ; INITIALIZE ADDRESS COUNTER
0260 8509               STA     ADDRCT          ; TO 16384
0262 A940               LDA     #16384/256
0264 850A               STA     ADDRCT+1
0266 20C202     STORPH: JSR     RAND            ; GENERATE A RANDOM NUMBER
0269 209402             JSR     MADDR           ; FORM A SCRAMBLED MEMORY ADDRESS
026C A505               LDA     RANDNO          ; STORE A RANDOM BYTE
026E A200               LDX     #0              ; INDIRECTLY THROUGH SCRAMBLED MEMORY
0270 8100               STA     (SCMEMA,X)      ; ADDRESS AT SCMEMA
0272 C609               DEC     ADDRCT          ; DECREMENT ADDRESS COUNTER
0274 D0F0               BNE     STORPH          ; AND LOOP IF NOT ZERO
0276 C60A               DEC     ADDRCT+1
0278 D0EC               BNE     STORPH
027A 60                 RTS                     ; RETURN WHEN DONE


                ;       RANDOM PATTERN STORED IN SCRAMBLED ORDER VERIFY ROUTINE
```

```
112 027B A940       RNDVER:  LDA    #16384/256   ; INITIALIZE ADDRESS COUNTER
113 027D 850A                STA    ADDRCT+1
114 027F 20C202     VERFPH:  JSR    RAND         ; GENERATE A RANDOM NUMBER
115 0282 209402              JSR    MADDR        ; FORM SCRAMBLED MEMORY ADDRESS
116 0285 A100                LDA    (SCMEMA,X)   ; GET DATA FROM MEMORY INDIRECTLY
117 0287 C505                CMP    RANDNO       ; THROUGH SCMEMA
118 0289 D008                BNE    VERRET       ; GO RETURN ON UNEQUAL COMPARE
119 028B C609                DEC    ADDRCT       ; DECREMENT ADDRESS COUNTER
120 028D D0F0                BNE    VERFPH       ; AND LOOP IF NOT ZERO
121 028F C60A                DEC    ADDRCT+1
122 0291 D0EC                BNE    VERFPH
123 0293 60         VERRET:  RTS                 ; RETURN
124
125                  ;       SCRAMBLED MEMORY ADDRESS FORMATION ROUTINE
126                  ;       USES ADDRCT AND SEED TO FORM A SCRAMBLED ADDRESS IN SCMEMA
127
128 0294 A507       MADDR:   LDA    SEED         ; GET LOWER BYTE OF RANDOM NUMBER
129 0296 4509                EOR    ADDRCT       ; EXCLUSIVE-OR WITH LOWER ADDRESS
130 0298 8500                STA    SCMEMA       ; LOWER BYTE OF RESULT
131 029A A508                LDA    SEED+1       ; GET UPPER BYTE OF RANDOM NUMBER
132 029C 450A                EOR    ADDRCT+1     ; EXCLUSIVE-OR WITH UPPER ADDRESS
133 029E 293F                AND    #16384/256-1 ; SAVE SIGNIFICANT BITS OF RESULT
134 02A0 C920                CMP    #8192/256    ; TEST IF IN USER RAM
135 02A2 B007                BCS    MADDR1       ; SKIP AHEAD IF NOT
136 02A4 18                  CLC
137 02A5 6D0002              ADC    USRRAM       ; IF SO, ADD IN FIRST PAGE ADDRESS OF USER
138 02A8 4CBF02              JMP    MADDR3       ; RAM
139 02AB C93F       MADDR1:  CMP    #16384-256/256 ; TEST IF IN SYSTEM RAM
140 02AD B00A                BCS    MADDR2       ; SKIP AHEAD IF NOT
141 02AF 38                  SEC
142 02B0 E920                SBC    #8192/256    ; IF SO, SUBTRACT OUT 8K AND THEN ADD IN
143 02B2 18                  CLC                 ; FIRST PAGE ADDRESS OF SYSTEM RAM
144 02B3 6D0102              ADC    SYSRAM
145 02B6 4CBF02              JMP    MADDR3
146 02B9 E93F       MADDR2:  SBC    #16384-256/256 ; MUST BE IN ROM AREA, SUBTRACT 15.75K
147 02BB 18                  CLC                 ; AND THEN ADD IN PAGE ADDRESS OF FILLER
148 02BC 6D0202              ADC    FILRAM       ; PAGE
149 02BF 8501       MADDR3:  STA    SCMEMA+1     ; STORE ADJUSTED PAGE ADDRESS
150 02C1 60                  RTS                 ; RETURN
151
152
153                  ;       RANDOM NUMBER GENERATOR SUBROUTINE
154                  ;       ENTER WITH SEED IN RANDNO
155                  ;       EXIT WITH NEW RANDOM NUMBER IN RANDNO
156                  ;       USES 16 BIT FEEDBACK SHIFT REGISTER METHOD
157                  ;       DESTROYS REGISTER A AND Y
158
159 02C2 A008        RAND:    LDY    #8           ; SET COUNTER FOR 8 RANDOM BITS
160 02C4 A505       RAND1:   LDA    RANDNO       ; EXCLUSIVE-OR BITS 3, 12, 14, AND 15
161 02C6 4A                  LSRA                ; OF SEED
162 02C7 4505                EOR    RANDNO
163 02C9 4A                  LSRA
164 02CA 4A                  LSRA
165 02CB 4505                EOR    RANDNO
166 02CD 4A                  LSRA
```



```
167 02CE 4506                EOR    RANDNO+1     ; RESULT IS IN BIT 3 OF A
168 02D0 4A                  LSRA                ; SHIFT INTO CARRY
169 02D1 4A                  LSRA
170 02D2 4A                  LSRA
171 02D3 4A                  LSRA
172 02D4 2606                ROL    RANDNO+1     ; SHIFT RANDNO LEFT ONE BRINGING IN CARRY
173 02D6 2605                ROL    RANDNO
174 02D8 88                  DEY                 ; TEST IF 8 NEW RANDOM BITS COMPUTED
175 02D9 D0E9                BNE    RAND1        ; LOOP FOR MORE IF NOT
176 02DB 60                  RTS                 ; RETURN
177
178 0000                     .END
NO ERROR LINES
```

DISK DRIVE DIAGNOSTIC PROGRAM LISTING

```
                        .PAGE  'DOCUMENTATION, EQUATES, AND STORAGE'
                ;       DISK DIAGNOSTIC FOR THE K-1013 FLOPPY DISK CONTROLLER
                ;       THIS PROGRAM HAS SEVERAL DIAGNOSTIC/EXERCISE ROUTINES THAT CAN
                ;       BE USED TO TROUBLESHOOT AND BURN IN K-1013 DISK CONTROL BOARDS.
                ;       THE ROUTINES ARE ORGAINZED INTO THREE "LEVELS".
                ;       LEVEL 1 TESTS THE BASIC I/O FUNCTION OF THE UPD765 CHIP AND THE
                ;       SEEK MECHANISM OF THE DISK DRIVE.
                ;       LEVEL 2 TESTS THE BASIC WRITE AND READ CAPABILITY OF THE DISK
                ;       DRIVE BY DOING A FORMAT (IN CODOS FORMAT) AND SECTOR BY SECTOR
                ;       READ WITH CRC CHECK.
                ;       LEVEL 3 THROROUGHLY TESTS THE READ AND WRITE CAPABILITY OF THE
                ;       DISK DRIVE BY WRITING RANDOM NUMBERS ON THE ENTIRE DISK AND THEN
                ;       READING THEM BACK FOR SOFTWARE VERIFICATION.
                ;       TO USE LEVEL X, ALL OF THE CODE FROM LEVEL 1 TO LEVEL X-1 MUST
                ;       ALSO BE ENTERED.
                ;
                ;       THE SPECIFIC TEST ROUTINES PROVIDED ARE AS FOLLOWS:
                ;
                ;       LEVEL 1     RECALIBRATE FOLLOWED BY 256 RANDOM SEEKS
                ;       LEVEL 2     FORMAT THE DISK IN CODOS FORMAT
                ;                   READ SECTORS SEQUENTIALLY CHECKING FOR CRC ERRORS
                ;       LEVEL 3     READ/WRITE TEST USING RANDOM NUMBERS
                ;
                ;       IN ADDITION THIS PROGRAM CONTAINS GENERALLY USEFUL SUBROUTINES
                ;       FOR OPERATING THE K-1013 DISK CONTROLLER.  THESE ARE AS FOLLOWS:
                ;
                ;       CMDPH     SEND STRING OF COMMAND BYTES TO THE UPD765
                ;       RSLPH     RECEIVE STRING OF STATUS BYTES FROM THE UPD765
                ;       DMASET    SET DMA ADDRESS REGISTER FROM PAGE ADDRESS IN A
                ;       SPECFY    SEND SPECIFY COMMAND
                ;       RECAL     SEND RECALIBRATE COMMAND
                ;       SEEK      SEEK TO SPECIFIC TRACK
                ;       READ      READ SPECIFIC SECTOR ON CURRENT TRACK
                ;       WRITE     WRITE SPECIFIC SECTOR ON CURRENT TRACK
                ;       RAND      GENERATE RANDOM NUMBER

                ;    ** EQUATES **

4000            USRRAM  =       X'4000          ; ADDRESS OF START OF USER RAM ON FDC
8000            SYSRAM  =       X'8000          ; ADDRESS OF START OF SYST RAM ON FDC
9FEE            FDCMSR  =       SYSRAM+X'1FEE ; DISK CONTROLLER MAIN STATUS REGISTER
9FEF            FDCDR   =       SYSRAM+X'1FEF ; DISK CONTROLLER DATA REGISTER
9FE8            FDCIRQ  =       SYSRAM+X'1FE8 ; DISK CONTROLLER INTERRUPT REQUEST FLAG
9FE8            FDCHWC  =       SYSRAM+X'1FE8 ; DISK CONTROLLER HARDWARE CONTROL WRITE
9FEA            FDCDMA  =       SYSRAM+X'1FEA ; DISK CONTROLLER DMA ADDRESS REGISTER
8000            BUFFER  =       SYSRAM          ; ADDRESS OF BUFFER AREA FOR ROUTINES
8000            READBF  =       BUFFER          ; BUFFER FOR SEQUENTIAL READ EXERCIZE
8000            FORMDA  =       BUFFER          ; BUFFER FOR DISK FORMAT
8000            RWTSDA  =       BUFFER          ; BUFFER FOR RANDOM READ/WRITE TEST
0008            NSTAG   =       8               ; NUMBER OF SECTORS TO STAGGER PER TRACK

                ;       PAGE ZERO STORAGE, NEED NOT BE INITIALIZED FROM TAPE

0000                    .=      0               ; START AT LOCATION 0
```



```
0000            ERRORC: .=.+    1               ; ERROR CODE
0001            DSKSTS: .=.+    8               ; ROOM FOR 8 STATUS BYTES FROM FDC
                ;*************************************************************************
                ;*******USER MUST SET DRIVE AND SIDE BEFORE RUNNING ANY TEST*************
                ;*************************************************************************
0009            DRIVE:  .=.+    1 ;*MUST SET*;  CURRENT DRIVE
000A            SIDE:   .=.+    1 ;*MUST SET*;  CURRENT SIDE (SET TO 0 FOR 1 SIDED DRIVE)
000B            TRACK:  .=.+    1               ; CURRENT TRACK NUMBER
000C            SECTOR: .=.+    1               ; CURRENT SECTOR NUMBER
000D            SEEKCT: .=.+    1               ; COUNT OF SUCCESSFUL RANDOM SEEKS
000E            STAGGR: .=.+    1               ; STAGGER COUNTER DURING FORMAT
000F            RANDNO: .=.+    2               ; RANDOM NUMBER REGISTER

                ;       LIST OF DISK COMMANDS, STARTS IN PROGRAM AREA BUT MUST BE IN
                ;       RAM SO THAT CERTAIN BYTES OF THE COMMANDS CAN BE CHANGED

0011                    .=      X'4000          ; START IN USER RAM ON K-1013

                ;/////////////BEGINNING OF LEVEL 1 DISK TEST CODE//////////////////////

                DSKCMD:                         ; START OF PREFORMATTED DISK COMMANDS
4000 03         SPECCM: .BYTE   X'03            ; SPECIFY COMMAND
4001 6F                 .BYTE   X'6F            ; SEEK SPEED=10MS  HEAD UNLOAD TIME=240MS
4002 24                 .BYTE   X'24            ; HEAD LOAD TIME=40MS  DMA MODE

4003 08         SNSICM: .BYTE   X'08            ; SENSE INTERRUPT STATUS COMMAND

4004 07         RECLCM: .BYTE   X'07            ; RECALIBRATE COMMAND
4005 00                 .BYTE   X'00            ; DRIVE NUMBER IN BITS 0-1

4006 0F         SEEKCM: .BYTE   X'0F            ; SEEK COMMAND
4007 00                 .BYTE   X'00            ; DRIVE NUMBER IN BITS 0-1, SIDE NUMBER B2
4008 00                 .BYTE   X'00            ; NEW CYLINDER NUMBER

4009 04         SSTSCM: .BYTE   X'04            ; SENSE DISK STATUS COMMAND
400A 00                 .BYTE   X'00            ; DRIVE NUMBER IN BITS 0-1, SIDE NUMBER B2

400B 46         READCM: .BYTE   X'46            ; READ DATA, MFM, READ DEL DATA
400C 00                 .BYTE   X'00            ; HEAD ZERO, DRIVE NUMBER IN BITS 0-1
400D 00                 .BYTE   X'00            ; NEEDS PRESENT CYLINDER NUMBER
400E 00                 .BYTE   X'00            ; NEEDS HEAD NUMBER
400F 00                 .BYTE   X'00            ; NEEDS SECTOR NUMBER TO READ
4010 01                 .BYTE   X'01            ; CODE FOR 256 BYTES PER SECTOR
4011 00                 .BYTE   X'00            ; LAST SECTOR TO READ
4012 0E                 .BYTE   X'0E            ; GAP LENGTH FOR 26 SECTORS, 256 BYTES/SECT
4013 FF                 .BYTE   X'FF            ; DATA LENGTH = X'FF SINCE N IS NON-ZERO

4014 4D         FORMCM: .BYTE   X'4D            ; FORMAT A DOUBLE DENSITY TRACK COMMAND
4015 00                 .BYTE   X'00            ; HEAD ZERO, DRIVE NUMBER IN BITS 0 AND 1
4016 01                 .BYTE   X'01            ; SPECIFY 256 BYTES/SECTOR
4017 1A                 .BYTE   X'1A            ; SPECIFY 26 SECTORS/TRACK
4018 36                 .BYTE   X'36            ; GAP LENGTH FOR 26 SECTORS, 256 BYTES/SECT
4019 EA                 .BYTE   X'EA            ; DATA FIELD FILLER BYTE

401A 45         WRITCM: .BYTE   X'45            ; WRITE DATA, MFM
```

```
                        .BYTE   X'00            ; HEAD ZERO, DRIVE NUMBER IN BITS 0-1
                        .BYTE   X'00            ; NEEDS PRESENT CYLINDER NUMBER
                        .BYTE   X'00            ; NEEDS HEAD NUMBER
                        .BYTE   X'00            ; NEEDS SECTOR NUMBER TO WRITE
                        .BYTE   X'01            ; CODE FOR 256 BYTES PER SECTOR
                        .BYTE   X'00            ; LAST SECTOR TO WRITE
                        .BYTE   X'0E            ; GAP LENGTH FOR 26 SECTORS, 256 BYTES/SEC
                        .BYTE   X'FF            ; DATA LENGTH = X'FF SINCE N IS NON-ZERO

                ;       MONITOR RETURN POINT - NORMALLY A BRK INSTRUCTION BUT CAN BE
                ;       CHANGED TO AN RTS OR JMP IF DESIRED

MONENT: .BYTE   0,0,0
```



```
                        .PAGE   'LEVEL 1 TEST ROUTINES'

                ;       FLOPPY DISK SEEK EXERCISE PROGRAM

SEEKE:  CLD                             ; INSURE BINARY ARITHMETIC
        LDA     #0                      ; CLEAR ERROR CODE
        STA     ERRORC
        JSR     SPECFY                  ; INITIALIZE THE DISK CONTROLLER
        BCS     ERROR1                  ; JUMP OUT IF ERROR
        LDA     DRIVE                   ; RECALIBRATE SELECTED DRIVE
        JSR     RECAL
        BCS     ERROR2                  ; JUMP OUT IF ERROR
        LDA     #0                      ; INITIALIZE 256 SEEK COUNT
        STA     SEEKCT
SEEKE0: JSR     RAND                    ; GET A RANDOM TRACK NUMBER
SEEKE1: CMP     #77                     ; MAKE SURE IT IS LESS THAN 77
        BCC     SEEKE2                  ; JUMP OUT IF LESS THAN 77
        SBC     #77                     ; SUBTRACT 77 IF NOT LESS
        JMP     SEEKE1                  ; AND CHECK AGAIN
SEEKE2: TAX                             ; SET AS ARGUMENT TO SEEK ROUTINE
        LDA     DRIVE                   ; SET DRIVE TO SEEK
        JSR     SEEK                    ; SEEK TO THE RANDOM TRACK
        BCS     ERROR3                  ; JUMP OUT IF UNSUCCESSFUL SEEK
        LDY     #20                     ; WAIT 20 MILLISECONDS FOR HEAD SETTLE
SEEKE3: LDX     #1000/5
SEEKE4: DEX
        BNE     SEEKE4
        DEY
        BNE     SEEKE3
        INC     SEEKCT                  ; INCREMENT SUCCESSFUL SEEK COUNT
        BNE     SEEKE0                  ; GO DO ANOTHER SEEK IF 256 NOT DONE
        LDX     #0                      ; DO A FINAL SEEK TO TRACK 0
        LDA     DRIVE                   ; SET DRIVE NUMBER TO SEEK
        JSR     SEEK                    ; SEEK TO TRACK 0
        BCS     ERROR4                  ; JUMP OUT IF TRACK 0 SENSE EARLY
        LDA     DRIVE                   ; DO A SENSE DRIVE STATUS TO DETERMINE IF
        STA     SSTSCM+1                ; REALLY ON TRACK 0
        LDX     #SSTSCM-DSKCMD
        LDY     #2
        JSR     CMDPH
        LDX     #0                      ; READ STATUS INTO DSKSTS
        JSR     RSLPH
        LDA     DSKSTS                  ; GET STATUS REG 3 CONTENTS
        AND     #X'10                   ; TEST TRACK 0 SENSE
        BEQ     ERROR4                  ; ERROR IF DID NOT REACH TRACK 0
        BNE     ERRORO                  ; JUMP IF NO ERROR

ERROR4: INC     ERRORC                  ; 04 FINAL SEEK TO TRACK 0 ERROR CODE
ERROR3: INC     ERRORC                  ; 03 INTERMEDIATE SEEK ERROR CODE
ERROR2: INC     ERRORC                  ; 02 RECALIBRATE ERROR CODE
ERROR1: INC     ERRORC                  ; 01 SPECIFY ERROR CODE
ERRORO: JMP     MONENT                  ; 00 NO ERROR, GO TO THE MONITOR


                ;       SEND COMMAND TO FLOPPY DISK CONTROLLER
```

```
                        ;       ENTER WITH RELATIVE ADDRESS OF COMMAND BYTES IN X (RELATIVE TO
                        ;       DSKCMD) AND NUMBER OF BYTES IN COMMAND IN Y.
                        ;       ROUTINE SENDS THE BYTES TO THE FLOPPY DISK CONTROLLER AND
                        ;       RETURNS WITH THE CARRY FLAG OFF.
                        ;       IF AN ERROR IS DETECTED, THE CARRY FLAG IS ON.
                        ;       IF THE CONTROLLER IS BUSY, THE ERROR RETURN IS TAKEN

408A ADEE9F     CMDPH:  LDA     FDCMSR          ; LOOK AT MAIN STATUS REGISTER
408D 2910               AND     #X'10           ; LOOK AT ALL OF THE BUSY BITS
408F D015               BNE     CMDPHE          ; ERROR IF EXECUTING PREVIOUS
                                                ; COMMAND (EXCEPT SEEK)

4091 ADEE9F     CMDPH1: LDA     FDCMSR          ; LOOK AT MAIN STATUS REGISTER
4094 10FB               BPL     CMDPH1          ; WAIT UNTIL REQUEST FOR MASTER GOES TRUE
4096 2940               AND     #X'40           ; TEST DATA DIRECTION BIT
4098 D00C               BNE     CMDPHE          ; ERROR IF FDC WANTS TO TALK
409A BD0040             LDA     DSKCMD,X        ; GET A COMMAND BYTE
409D 8DEF9F             STA     FDCDR           ; STORE IT IN THE DISK CONTROLLER
40A0 E8                 INX                     ; POINT X TO NEXT COMMAND BYTE
40A1 88                 DEY                     ; DECREMENT COMMAND BYTE COUNT
40A2 D0ED               BNE     CMDPH1          ; GO TRANSFER NEXT BYTE IF NOT DONE
40A4 18                 CLC                     ; CLEAR CARRY FOR NORMAL RETURN
40A5 60                 RTS

40A6 38         CMDPHE: SEC                     ; SET CARRY FOR ERROR RETURN
40A7 60                 RTS


                        ;       RECEIVE STATUS FROM FLOPPY DISK CONTROLLER
                        ;       STATUS BYTES ARE STORED SEQUENTIALLY IN MEMORY STARTING AT
                        ;       DSKSTS+(X), THE NUMBER READ IS DETERMINED BY THE FDC BUSY STATUS
                        ;       ROUTINE READS THE STATUS BYTES IN THE RESULT PHASE AND RETURNS
                        ;       WITH THE CARRY FLAG OFF.
                        ;       IF AN ERROR IS DETECTED, THE CARRY FLAG IS ON

40A8 ADEE9F     RSLPH:  LDA     FDCMSR          ; LOOK AT MAIN STATUS REGISTER
40AB 10FB               BPL     RSLPH           ; WAIT UNTIL REQUEST FOR MASTER GOES TRUE
40AD 2940               AND     #X'40           ; TEST DATA DIRECTION BIT
40AF F011               BEQ     RSLPHE          ; ERROR IF FDC WANTS TO LISTEN
40B1 ADEF9F             LDA     FDCDR           ; GET A STATUS BYTE FROM THE DATA REGISTER
40B4 9501               STA     DSKSTS,X        ; PUT IT INTO MEMORY
40B6 E8                 INX                     ; POINT X TO NEXT STATUS BYTE
40B7 EA                 NOP                     ; DAMN SLOW CONTROLLER CHIP! 12US RESPONSE
40B8 EA                 NOP                     ; TIME FROM READ TO VALID BUSY STATUS
40B9 A910               LDA     #X'10           ; LOOK AT BUSY BIT IN MAIN STATUS REGISTER
40BB 2DEE9F             AND     FDCMSR
40BE D0E8               BNE     RSLPH           ; GO FOR ANOTHER STATUS BYTE IF STILL BUSY
40C0 18                 CLC                     ; CLEAR CARRY FOR NORMAL RETURN
40C1 60                 RTS

40C2 38         RSLPHE: SEC                     ; SET CARRY FOR ERROR RETURN
40C3 60                 RTS


                        ;       FLOPPY DISK RESET AND SPECIFY.  THIS ROUTINE FIRST RESETS THE
```



```
                        ;       FLOPPY DISK CONTROLLER AND THEN ESTABLISHES THE SEEK RATE, HEAD
                        ;       LOAD TIME, HEAD UNLOAD TIME, AND DMA OPTION BY USING A SPECIFY
                        ;       COMMAND.  SPECIFY DATA MUST BE PREFORMATTED IN MEMORY AT SPECDA.
                        ;       NORMAL RETURN WITH CARRY OFF, ERROR RETURN WITH CARRY ON NO
                        ;       STATUS IS GENERATED SO THE STATUS BYTES IN DSKSTS ARE UNCHANGED.

40C4 ADE89F     SPECFY: LDA     FDCIRQ          ; TEST IF AN INTERRUPT IS PENDING
40C7 300C               BMI     SPECF1          ; JUMP AHEAD IF NOT
40C9 A203               LDX     #SNSICM-DSKCMD; EXECUTE SENSE INTERRUPT STATUS COMMAND
40CB A001               LDY     #1
40CD 208A40             JSR     CMDPH
40D0 A200               LDX     #0              ; READ STATUS BYTES INTO DSKSTS
40D2 20A840             JSR     RSLPH

40D5 A200       SPECF1: LDX     #SPECCM-DSKCMD; SET ADDRESS OF SPECIFY COMMAND
40D7 A003               LDY     #3              ; THREE BYTES IN SPECIFY COMMAND
40D9 4C8A40             JMP     CMDPH           ; EXECUTE THE SPECIFY COMMAND
                                                ; AND RETURN


                        ;       FLOPPY DISK RECALIBRATE
                        ;       THIS ROUTINE SEEKS THE DRIVE NUMBER LOADED INTO A TO TRACK ZERO
                        ;       REGARDLESS OF WHERE IT CURRENTLY IS.
                        ;       THE STATUS BYTES AT DSKSTS ARE MODIFIED AS A RESULT OF THE
                        ;       RECALIBRATE OPERATION.
                        ;       IN ADDITION THEY ARE CHECKED AND THE CARRY SET IF FOR ANY REASON
                        ;       THE OPERATION WAS NOT SUCCESSFULLY PERFORMED.

                        ;       RECALIBRATION IS PERFORMED BY ISSUEING A RECALIBRATE COMMAND,
                        ;       WAITING FOR INTERRUPT REQUEST TO GO TRUE, AND PERFORMING A
                        ;       SENSE INTERRUPT STATUS COMMAND.

40DC 2903       RECAL:  AND     #X'03           ; FORMAT DRIVE NUMBER INTO SECOND BYTE OF
40DE 8D0540             STA     RECLCM+1        ; RECALIBRATE COMMAND
40E1 A204               LDX     #RECLCM-DSKCMD; EXECUTE THE RECALIBRATE COMMAND
40E3 A002               LDY     #2              ; 2 BYTES IN RECALIBRATE COMMAND
40E5 208A40             JSR     CMDPH
40E8 B01F               BCS     RECALE          ; JUMP OUT IF ERROR

40EA ADE89F     RECAL1: LDA     FDCIRQ          ; READ DISK CONTROLLER INTERRUPT REQUEST
                                                ; (FDC IRQ JUMPER REMOVED OR INTERRUPTS
                                                ; DISABLED
40ED 30FB               BMI     RECAL1          ; WAIT UNTIL THE FDC REQUESTS AN INTERRUPT

40EF A203               LDX     #SNSICM-DSKCMD; EXECUTE SENSE INTERRUPT STATUS COMMAND
40F1 A001               LDY     #1              ; ONE BYTE IN SENSE INTERRUPT STATUS
40F3 208A40             JSR     CMDPH

40F6 A200               LDX     #0              ; SET UP TO READ STATUS INTO DSKSTS
40F8 20A840             JSR     RSLPH

40FB A502               LDA     DSKSTS+1        ; LOOK AT PRESENT CYLINDER NUMBER
40FD D00A               BNE     RECALE          ; ERROR IF NON-ZERO
40FF A501               LDA     DSKSTS          ; LOOK AT STO STATUS REGISTER
4101 29F8               AND     #X'F8           ; DELETE DON'T CARE BITS
```

```
290 4103 C920            CMP    #X'20          ; REMAINING BITS MUST BE X'20
291 4105 D002            BNE    RECALE         ; GO TO ERROR RETURN
292 4107 18              CLC                   ; CLEAR CARRY FOR NORMAL RETURN
293 4108 60              RTS
294
295 4109 38      RECALE: SEC                   ; SET CARRY FOR ERROR RETURN
296 410A 60              RTS
297
298
299              ;       FLOPPY DISK SEEK
300              ;       THIS ROUTINE SEEKS THE DRIVE NUMBER LOADED INTO A TO THE TRACK
301              ;       NUMBER LOADED INTO X.
302              ;       THE STATUS BYTES AT DSKSTS ARE MODIFIED AS A RESULT OF THE
303              ;       SEEK OPERATION.
304              ;       IN ADDITION THEY ARE CHECKED AND THE CARRY SET IF FOR ANY REASON
305              ;       THE OPERATION WAS NOT SUCCESSFULLY PERFORMED.
306
307              ;       SEEK IS PERFORMED BY ISSUEING A SEEK COMMAND,
308              ;       WAITING FOR INTERRUPT REQUEST TO GO TRUE, AND PERFORMING A
309              ;       SENSE INTERRUPT STATUS COMMAND.
310
311 410B 2903      SEEK:  AND    #X'03          ; FORMAT DRIVE NUMBER INTO SECOND BYTE OF
312 410D 8D0740          STA    SEEKCM+1       ; SEEK COMMAND AND SELECT HEAD 0
313 4110 8E0840          STX    SEEKCM+2       ; STORE NEW CYLINDER NUMBER INTO THIRD BYTE
314 4113 8A              TXA                   ; SAVE NEW CYLINDER NUMBER ON THE STACK
315 4114 48              PHA
316 4115 A206            LDX    #SEEKCM-DSKCMD; EXECUTE THE SEEK COMMAND
317 4117 A003            LDY    #3             ; 2 BYTES IN SEEK COMMAND
318 4119 208A40          JSR    CMDPH
319 411C B020            BCS    SEEKER         ; JUMP OUT IF ERROR
320
321 411E ADE89F  SEEK1: LDA     FDCIRQ         ; READ DISK CONTROLLER INTERRUPT REQUEST
322                              ;              (FDC IRQ JUMPER REMOVED OR INTERRUPTS
323                              ;              DISABLED
324 4121 30FB            BMI    SEEK1          ; WAIT UNTIL THE FDC REQUESTS AN INTERRUPT
325
326 4123 A203            LDX    #SNSICM-DSKCMD; EXECUTE SENSE INTERRUPT STATUS COMMAND
327 4125 A001            LDY    #1             ; ONE BYTE IN SENSE INTERRUPT STATUS
328 4127 208A40          JSR    CMDPH
329
330 412A A200            LDX    #0             ; SET UP TO READ STATUS INTO DSKSTS
331 412C 20A840          JSR    RSLPH
332
333 412F 68              PLA                   ; COMPARE PRESENT CYLINDER NUMBER WITH
334 4130 C502            CMP    DSKSTS+1       ; DESIRED CYLINDER NUMBER
335 4132 D00A            BNE    SEEKER         ; ERROR IF NOT THE SAME
336 4134 A501            LDA    DSKSTS         ; LOOK AT STO STATUS REGISTER
337 4136 29F8            AND    #X'F8          ; DELETE DON'T CARE BITS
338 4138 C920            CMP    #X'20          ; REMAINING BITS MUST BE X'20
339 413A D002            BNE    SEEKER         ; GO TO ERROR RETURN
340 413C 18              CLC                   ; CLEAR CARRY FOR NORMAL RETURN
341 413D 60              RTS
342
343 413E 38      SEEKER: SEC                   ; SET CARRY FOR ERROR RETURN
344 413F 60              RTS
```



```
345
346
347              ;       RANDOM NUMBER GENERATOR SUBROUTINE
348              ;       ENTER WITH SEED IN RANDNO
349              ;       EXIT WITH NEW RANDOM NUMBER IN RANDNO AND A
350              ;       USES 16 BIT FEEDBACK SHIFT REGISTER METHOD
351              ;       DESTROYS REGISTER A AND Y
352
353 4140 A008      RAND:  LDY    #8             ; SET COUNTER FOR 8 RANDOM BITS
354 4142 A50F      RAND1: LDA    RANDNO         ; EXCLUSIVE-OR BITS 3, 12, 14, AND 15
355 4144 4A              LSRA                  ; OF SEED
356 4145 450F            EOR    RANDNO
357 4147 4A              LSRA
358 4148 4A              LSRA
359 4149 450F            EOR    RANDNO
360 414B 4A              LSRA
361 414C 4510            EOR    RANDNO+1       ; RESULT IS IN BIT 3 OF A
362 414E 4A              LSRA                  ; SHIFT INTO CARRY
363 414F 4A              LSRA
364 4150 4A              LSRA
365 4151 4A              LSRA
366 4152 2610            ROL    RANDNO+1       ; SHIFT RANDNO LEFT ONE BRINGING IN CARRY
367 4154 260F            ROL    RANDNO
368 4156 88              DEY                   ; TEST IF 8 NEW RANDOM BITS COMPUTED
369 4157 D0E9            BNE    RAND1          ; LOOP FOR MORE IF NOT
370 4159 A50F            LDA    RANDNO
371 415B 60              RTS                   ; RETURN
372
373
374              ;//////////////END OF LEVEL 1 TEST ROUTINES//////////////////////////
375
```

```
                        .PAGE  'LEVEL 2 DISK TEST ROUTINES'
376
377             ;//////////////BEGINNING OF LEVEL 2 DISK TEST ROUTINES//////////////////
378
379             ;       FORMAT A DISK FOR DOUBLE DENSITY, 26 SECTORS OF 256 BYTES,
380             ;       SECTORS ARE ALTERNATELY NUMBERED AND ARE STAGGERED BY 8 SECTORS
381             ;       FROM TRACK-TO-TRACK.
382
383 415C D8        FORMAT: CLD                     ; INSURE BINARY ARITHMETIC
384 415D A910           LDA     #X'10           ; INITIALIZE TO NO ERROR
385 415F 8500           STA     ERRORC
386 4161 20C440         JSR     SPECFY          ; INITIALIZE THE DISK CONTROLLER
387 4164 9003           BCC     FORMY
388 4166 4C0542         JMP     FRMER1          ; JUMP OUT IF ERROR
389 4169 A509   FORMY:  LDA     DRIVE           ; RECALIBRATE DESIRED DRIVE
390 416B 20DC40         JSR     RECAL
391 416E 9003           BCC     FORMZ
392 4170 4C0342         JMP     FRMER2          ; JUMP OUT IF ERROR
393 4173 A900   FORMZ:  LDA     #0              ; SET DMA READ MODE, UNPROTECT SYSRAM
394 4175 8DE89F         STA     FDCHWC
395 4178 A900           LDA     #0              ; INITIALIZE TRACK NUMBER
396 417A 850B           STA     TRACK
397 417C 850E           STA     STAGGR          ; INITIALIZE SECTOR STAGGER COUNT
398
399             ;       CREATE FORMATTING DATA AREA FOR THE NEXT TRACK
400
401 417E A60E   FORM1:  LDX     STAGGR          ; INITIALIZE SECTOR NUMBER POINTER
402 4180 A000           LDY     #0              ; INITIALIZE POINTER INTO FORMAT DATA AREA
403 4182 A50B   FORM2:  LDA     TRACK           ; PUT TRACK #,
404 4184 990080         STA     FORMDA,Y
405 4187 A50A           LDA     SIDE            ; SIDE #,
406 4189 C8             INY
407 418A 990080         STA     FORMDA,Y
408 418D C8             INY
409 418E BD1442         LDA     ALTTAB,X        ; LOOK UP SECTOR NUMBER IN ALTERNATE TABLE
410 4191 990080         STA     FORMDA,Y
411 4194 AD1640         LDA     FORMCM+2        ; NUMBER OF BYTES PER SECTOR
412 4197 C8             INY
413 4198 990080         STA     FORMDA,Y        ; INTO FORMAT DATA AREA IN MEMORY
414 419B C8             INY
415 419C E8             INX                     ; INCREMENT SECTOR NUMBER POINTER
416 419D 8A             TXA                     ; MOD 26
417 419E 200A42         JSR     MOD26
418 41A1 AA             TAX
419 41A2 C068           CPY     #26*4           ; TEST IF 26 SECTORS GONE THROUGH
420 41A4 D0DC           BNE     FORM2           ; CONTINUE GENERATING DATA IF NOT
421
422             ;       SET UP THE FORMAT COMMAND
423
424 41A6 A980           LDA     #FORMDA/256     ; SET DMA ADDRESS FOR FORMATTING DATA
425 41A8 208142         JSR     DMASET          ; CALCULATE CORRESPONDING DMA ADDRESS
426                                             ; AND PUT INTO DMA ADDRESS REGISTER
427 41AB A50A           LDA     SIDE            ; SET UP SECOND BYTE OF FORMAT COMMAND
428 41AD 0A             ASLA                    ; ACCORDING TO DRIVE NUMBER AND SIDE
429 41AE 0A             ASLA
```



```
430 41AF 0509           ORA     DRIVE
431 41B1 8D1540         STA     FORMCM+1
432 41B4 A214           LDX     #FORMCM-DSKCMD
433 41B6 A006           LDY     #6              ; 6 BYTES IN FORMAT COMMAND
434 41B8 208A40         JSR     CMDPH           ; SEND THE COMMAND TO THE DISK CONTROLLER
435 41BB B044           BCS     FRMER3          ; COMMAND PHASE ERROR EXIT
436 41BD ADE89F FORM3:  LDA     FDCIRQ          ; WAIT UNTIL INTERRUPT REQUEST FROM FDC
437 41C0 30FB           BMI     FORM3
438 41C2 A200           LDX     #0              ; SET UP TO READ STATUS INTO STATUS AREA
439 41C4 20A840         JSR     RSLPH
440 41C7 B036           BCS     FRMER4          ; JUMP OUT ON GROSS ERROR
441 41C9 A501           LDA     DSKSTS          ; CHECK STATUS REGISTER 0
442 41CB 29D8           AND     #X'D8           ; MASK OUT NON-ERROR BITS
443 41CD D030           BNE     FRMER4          ; GO TO ERROR IF ANY OF REMAINDER SET
444 41CF A502           LDA     DSKSTS+1        ; CHECK STATUS REGISTER 1
445 41D1 2935           AND     #X'35           ; MASK OUT NON-ERROR BITS
446 41D3 D02A           BNE     FRMER4          ; GO TO ERROR IF ANY OF REMAINDER SET
447 41D5 A503           LDA     DSKSTS+2        ; CHECK STATUS REGISTER 2
448 41D7 2933           AND     #X'33           ; MASK OUT NON-ERROR BITS
449 41D9 D024           BNE     FRMER4          ; GO TO ERROR IF ANY OF REMAINDER SET
450
451 41DB A000           LDY     #0              ; WAIT AWHILE BEFORE SEEKING TO PREVENT
452 41DD 88     FORM4:  DEY                     ; OVERLAP WITH ERASE CURRENT ON DISK DRIVE
453 41DE D0FD           BNE     FORM4
454 41E0 A50E           LDA     STAGGR          ; ADD IN STAGGER FACTOR FOR NEXT TRACK
455 41E2 18             CLC
456 41E3 6912           ADC     #26-NSTAG
457 41E5 200A42         JSR     MOD26           ; SUM MOD 26
458 41E8 850E           STA     STAGGR          ;
459 41EA E60B           INC     TRACK           ; INCREMENT TRACK NUMBER
460 41EC A50B           LDA     TRACK
461 41EE C94D           CMP     #77             ; TEST IF ALL TRACKS FORMATTED
462 41F0 F015           BEQ     FRMER0          ; GO TO MONITOR IF SO, FINISHED
463 41F2 AA             TAX                     ; SEEK TO THE NEW TRACK
464 41F3 A509           LDA     DRIVE           ; ON SPECIFIED DRIVE
465 41F5 200B41         JSR     SEEK
466 41F8 B003           BCS     FRMER5          ; JUMP OUT ON SEEK ERROR
467 41FA 4C7E41         JMP     FORM1           ; GO FORMAT THE NEXT TRACK
468
469 41FD E600   FRMER5: INC     ERRORC          ; 15 SET SEEK ERROR CODE
470 41FF E600   FRMER4: INC     ERRORC          ; 14 SET FORMAT RESULT PHASE ERROR CODE
471 4201 E600   FRMER3: INC     ERRORC          ; 13 SET FORMAT COMMAND PHASE ERROR CODE
472 4203 E600   FRMER2: INC     ERRORC          ; 12 SET RECALIBRATE ERROR CODE
473 4205 E600   FRMER1: INC     ERRORC          ; 11 SET SPECIFY ERROR CODE
474 4207 4C2340 FRMER0: JMP     MONENT          ; 10 NO ERROR, RETURN TO MONITOR
475
476 420A C91A   MOD26:  CMP     #26             ; RETURN A MOD 26
477 420C 9005           BCC     MOD26A          ; SKIP IF OK
478 420E E91A           SBC     #26             ; IF NOT, SUBTRACT 26
479 4210 4C0A42         JMP     MOD26           ; AND TRY AGAIN
480 4213 60     MOD26A: RTS                     ; RETURN
481
482 4214 000D010E ALTTAB: .BYTE  0,13,1,14       ; TABLE OF INTERLEAVED SECTOR NUMBERS
483 4218 020F0310         .BYTE  2,15,3,16       ; INTERLEAVE FACTOR = 2
484 421C 04110512         .BYTE  4,17,5,18
```

```
4220 06130714          .BYTE  6,19,7,20
4224 08150916          .BYTE  8,21,9,22
4228 0A170B18          .BYTE  10,23,11,24
422C 0C19              .BYTE  12,25


                ;      SEQUENTIAL READ EXERCIZER
                ;      READS DISK 0 ONE SECTOR AT A TIME FROM TRACK 0 SECTOR 0 TO
                ;      TRACK 76 SECTOR 25 AND THEN REPEATS.

422E D8         READEX: CLD                    ; INSURE BINARY ARITHMETIC
422F A920               LDA    #X'20           ; INITIALIZE ERROR CODE
4231 8500               STA    ERRORC
4233 20C440             JSR    SPECFY          ; INITIALIZE DISK CONTROLLER
4236 B044               BCS    RERR1           ; JUMP OUT IF ERROR
4238 A509       REX1:   LDA    DRIVE           ; RECALIBRATE DESIGNATED DRIVE
423A 20DC40             JSR    RECAL
423D B03B               BCS    RERR2           ; JUMP OUT IF ERROR
423F A900               LDA    #0              ; SET TRACK TO ZERO
4241 850B               STA    TRACK
4243 A900       REX2:   LDA    #0              ; SET SECTOR TO ZERO
4245 850C               STA    SECTOR
4247 A50B       REX3:   LDA    TRACK           ; ESTABLISH TRACK,
4249 8D0840             STA    SEEKCM+2
424C A60C               LDX    SECTOR          ; SECTOR,
424E A080               LDY    #READBF/256     ; MEMORY ADDRESS,
4250 A50A               LDA    SIDE            ; AND SIDE/DRIVE NUMBER
4252 0A                 ASLA
4253 0A                 ASLA
4254 0509               ORA    DRIVE
4256 20B342             JSR    READ            ; READ THE SECTOR INTO MEMORY
4259 B01D               BCS    RERR3           ; JUMP OUT IF READ ERROR
425B E60C               INC    SECTOR          ; INCREMENT SECTOR NUMBER
425D A50C               LDA    SECTOR
425F C91A               CMP    #26             ; TEST IF ALL SECTORS READ
4261 D0E4               BNE    REX3            ; GO READ NEXT SECTOR IF NOT
4263 E60B               INC    TRACK           ; INCREMENT TRACK NUMBER
4265 A50B               LDA    TRACK
4267 C94D               CMP    #77             ; TEST IF ALL TRACKS READ
4269 F0CD               BEQ    REX1            ; GO READ DISK AGAIN IF SO
426B AA                 TAX                    ; SEEK TO THE NEW TRACK
426C A509               LDA    DRIVE           ; ON DESIGNATED DRIVE
426E 200B41             JSR    SEEK
4271 B003               BCS    RERR4           ; JUMP OUT ON SEEK ERROR
4273 4C4342             JMP    REX2            ; GO READ SECTOR 0 ON NEW TRACK


4276 E600       RERR4:  INC    ERRORC          ; 24 SET SEEK ERROR CODE
4278 E600       RERR3:  INC    ERRORC          ; 23 SET READ ERROR CODE
427A E600       RERR2:  INC    ERRORC          ; 22 SET RECALIBRATE ERROR CODE
427C E600       RERR1:  INC    ERRORC          ; 21 SET SPECIFY ERROR CODE
427E 4C2340     RERR0:  JMP    MONENT          ; 20 NO ERROR, RETURN TOTHE MONITOR


                ;      SUBROUTINE TO ACCEPT THE PAGE ADDRESS OF A MEMORY BUFFER AND
```



```
                ;      SET THE DMA ADDRESS REGISTER WITH THE APPROPRIATE VALUE.  THE
                ;      BUFFER ADDRESS MUST BE IN THE DISK CONTROLLER RAM.
                ;      RETURN WITH CARRY CLEAR IF ADDRESS IS OK, SET IF ADDRESS IS
                ;      INVALID.
                ;      FOR THIS ROUTINE TO WORK, THE ORIGIN OF THE USER RAM MUST BE
                ;      EQUATED TO USRRAM AND THE ORIGIN OF THE SYSTEM RAM MUST BE
                ;      EQUATED TO SYSRAM.
                ;      ENTER WITH BUFFER PAGE ADDRESS IN A, EXIT WITH DMA ADDRESS IN A
                ;      AND THE DMA ADDRESS REGISTER.

4281 C940       DMASET: CMP    #USRRAM/256     ; COMPARE WITH BEGINNING OF USER RAM
4283 900E               BCC    DMAST1          ; JUMP IF LESS THAN USER RAM
4285 C960               CMP    #USRRAM/256+32  ;COMPARE WITH END OF USER RAM
4287 B00A               BCS    DMAST1          ; JUMP IF NOT IN USER RAM
4289 38                 SEC                    ; IF IN USER RAM, COMPUTE RELATIVE PAGE
428A E940               SBC    #USRRAM/256     ; ADDRESS IN USER RAM
428C 48                 PHA                    ; SAVE RESULT
428D A900               LDA    #USRRAM/32      ; TEST IF USRRAM IS ON AN ODD 4K BOUNDARY
428F 3012               BMI    DMAST2          ; JUMP ON ODD BOUNDARY
4291 1016               BPL    DMAST3          ; JUMP ON EVEN BOUNDARY
4293 C980       DMAST1: CMP    #SYSRAM/256     ; COMPARE WITH BEGINNING OF SYSTEM RAM
4295 901A               BCC    DMASTE          ; ERROR IF NOT IN DISK CONTROLLER RAM
4297 C99F               CMP    #SYSRAM/256+31  ;COMPARE WITH END OF USABLE SYSTEM RAM
4299 B016               BCS    DMASTE          ; ERROR IF NOT IN DISK CONTROLLER RAM
429B 38                 SEC                    ; IF IN SYSTEM RAM, COMPUTE RELATIVE PAGE
429C E960               SBC    #SYSRAM/256-32  ;ADDRESS IN SYSTEM RAM
429E 48                 PHA                    ; SAVE RESULT
429F A900               LDA    #SYSRAM/32      ; TEST IF SYSRAM IS ON AN ODD 4K BOUNDARY
42A1 1006               BPL    DMAST3          ; JUMP ON EVEN BOUNDARY
42A3 68         DMAST2: PLA                    ; RETRIEVE RELATIVE ADDRESS
42A4 4910               EOR    #X'10           ; FLIP BIT 6 (AFTER SHIFT) FOR ODD BOUNDARY
42A6 4CAA42             JMP    DMAST4
42A9 68         DMAST3: PLA                    ; RETRIEVE RELATIVE ADDRESS
42AA 0A         DMAST4: ASLA                   ; MULTIPLY RELATIVE PAGE ADDRESS BY 4
42AB 0A                 ASLA
42AC 8DEA9F             STA    FDCDMA          ; AND PUT RESULT IN DMA ADDRESS REGISTER
42AF 18                 CLC
42B0 60                 RTS                    ; NORMAL RETURN
42B1 38         DMASTE: SEC                    ; ERROR RETURN
42B2 60                 RTS


                ;      FLOPPY DISK READ SECTOR
                ;      THIS ROUTINE READS ONE SECTOR FROM DISK INTO MEMORY.
                ;      ENTER WITH MEMORY PAGE NUMBER TO RECEIVE SECTOR IN Y (MUST BE
                ;      IN DISK CONTROLLER RAM), SECTOR NUMBER TO READ IN X (READS
                ;      CURRENT TRACK), AND DISK DRIVE NUMBER AND DISKETTE SIDE IN A.
                ;      ***CAUTION-CURRENT CYLINDER NUMBER IS TAKEN FROM SEEKCM+2, IF
                ;      IT DOES NOT MATCH THE ACTUAL HEAD POSITION, A READ ERROR WILL
                ;      RESULT***
                ;      RETURNS WITH CARRY CLEAR IF NO READ ERROR, CARRY SET IF A READ
                ;      OR OTHER KIND OF ERROR.
                ;      THE STATUS AREA IS UPDATED AFTER A READ, 7 BYTES ARE STORED.

                ;      NOTE: USES DMA MODE BUT WAITS FOR OPERATION TO COMPLETE BEFORE
```

```
                        ;       RETURNING.
                        ;       A READ DATA COMMAND IS USED  AND THE SK BIT IS ZERO.  THIS
                        ;       MEANS THAT DELETED SECTORS WILL BE READ AND BIT 6 OF STATUS 2
                        ;       WILL BE SET ON RETURN.
                        ;
600 42B3 8D0C40 READ:   STA     READCM+1        ; PUT DRIVE/SIDE IN SECOND BYTE OF COMMAND
601 42B6 4A             LSRA                    ; ISOLATE HAED # AND PUT INTO H FIELD
602 42B7 4A             LSRA
603 42B8 8D0E40         STA     READCM+3
604 42BB AD0840         LDA     SEEKCM+2        ; ESTABLISH CYLINDER NUMBER
605 42BE 8D0D40         STA     READCM+2
606 42C1 8A             TXA                     ; GET SECTOR NUMBER
607 42C2 8D0F40         STA     READCM+4        ; SET AS SECTOR TO READ
608 42C5 8D1140         STA     READCM+6        ; SET ALSO AS LAST SECTOR TO READ
609 42C8 98             TYA                     ; GET MEMORY PAGE ADDRESS
610 42C9 208142         JSR     DMASET          ; SET DMA ADDRESS REGISTER ACCORDINGLY
611 42CC A901           LDA     #X'01           ; SET DMA WRITE MODE AND SYSTEM RAM WRITE
612 42CE 8DE89F         STA     FDCHWC          ; ALLOW
613 42D1 A009           LDY     #9              ; 9 BYTES IN READ COMMAND
614 42D3 A20B           LDX     #READCM-DSKCMD; RELATIVE ADDRESS OF READ COMMAND
615 42D5 208A40         JSR     CMDPH           ; SEND COMMAND TO DISK CONTROLLER
616 42D8 B028           BCS     READER          ; JUMP OUT ON ERROR
617
618 42DA ADE89F READ3:  LDA     FDCIRQ          ; WAIT UNTIL INTERRUPT REQUEST FROM FDC
619 42DD 30FB           BMI     READ3
620
621 42DF A200           LDX     #0              ; SET UP TO READ STATUS INTO STATUS AREA
622 42E1 20A840         JSR     RSLPH
623 42E4 B01C           BCS     READER          ; JUMP OUT ON GROSS ERROR
624 42E6 A501           LDA     DSKSTS          ; CHECK STATUS REGISTER 0
625 42E8 29D8           AND     #X'D8           ; MASK OUT NON-ERROR BITS
626 42EA F008           BEQ     READ4           ; JUMP AHEAD IF NO OBVIOUS ERROR
627 42EC C940           CMP     #X'40           ; TEST IF ABNORMAL TERMINATION ERROR
628 42EE D012           BNE     READER          ; TRUE ERROR IF NOT
629 42F0 A502           LDA     DSKSTS+1        ; IF ABNORMAL TERMINATION, TEST IF END OF
630 42F2 100E           BPL     READER          ; CYLINDER, TRUE ERROR IF NOT, OK IF SO
631 42F4 A502 READ4:    LDA     DSKSTS+1        ; CHECK STATUS REGISTER 1
632 42F6 2935           AND     #X'35           ; MASK OUT NON-ERROR BITS
633 42F8 D008           BNE     READER          ; GO TO ERROR IF ANY OF REMAINDER SET
634 42FA A503           LDA     DSKSTS+2        ; CHECK STATUS REGISTER 2
635 42FC 2933           AND     #X'33           ; MASK OUT NON-ERROR BITS
636 42FE D002           BNE     READER          ; GO TO ERROR IF ANY OF REMAINDER SET
637
638 4300 18             CLC                     ; CLEAR CARRY FOR NORMAL RETURN
639 4301 60             RTS
640 4302 38 READER:     SEC                     ; SET CARRY IF ANY ERROR DETECTED
641 4303 60             RTS                     ; AND RETURN
642
643                     ;//////////////END OF LEVEL 2 ROUTINES////////////////////////////////
644
```



```
                        .PAGE  'RANDOM DATA WRITE AND READ EXERCIZE'
645                     ;       RANDOM DATA WRITE AND READ EXERCIZE
646                     ;       RANDOM DATA IS WRITTEN INTO ALL TRACKS AND SECTORS OF DISK 0
647                     ;       AND THEN READ BACK AND COMPARED WITH THE ORIGINAL
648                     ;       RANDOM SEQUENCE.  ANY ERROR WILL TERMINATE THE PROGRAM.
649                     ;       DISK FORMAT ASSUMED TO BE 26 SECTORS OF 256 BYTES EACH, DOUBLE
650                     ;       DENSITY.
651
652                     ;//////////////BEGINNING OF LEVEL 3////////////////////////////////
653
654 4304 D8 RWTST:      CLD                     ; INSURE BINARY ARITHMETIC
655 4305 A930           LDA     #X'30           ; CLEAR ERROR CODE FOR READ/WRITE TEST
656 4307 8500           STA     ERRORC
657 4309 A900           LDA     #0              ; ALLOW WRITING INTO SYSTEM RAM
658 430B 8DE89F         STA     FDCHWC
659 430E A901           LDA     #X'01           ; INITIALIZE RANDOM NUMBER SEED
660 4310 850F           STA     RANDNO
661 4312 8510           STA     RANDNO+1
662 4314 20C440         JSR     SPECFY          ; INITIALIZE DISK CONTROLLER
663 4317 9003           BCC     RWTSTZ
664 4319 4CCA43         JMP     RWERR1          ; JUMP OUT IF ERROR
665 431C A509 RWTSTZ:   LDA     DRIVE           ; RECALIBRATE DESIGNATED DRIVE
666 431E 20DC40         JSR     RECAL
667 4321 9003           BCC     RWTSTY
668 4323 4CC843         JMP     RWERR2          ; JUMP OUT IF ERROR
669 4326 A900 RWTSTY:   LDA     #0              ; SET TRACK NUMBER TO ZERO
670 4328 850B           STA     TRACK
671 432A A900 RWTST1:   LDA     #0              ; SET SECTOR NUMBER TO ZERO
672 432C 850C           STA     SECTOR
673 432E A200 RWTST2:   LDX     #0              ; INITIALIZE DATA POINTER FOR RANDOM DATA
674 4330 204041 RWTST3: JSR     RAND            ; GET A RANDOM DATA BYTE
675 4333 9D0080         STA     RWTSDA,X        ; STORE IT IN THE BUFFER
676 4336 E8             INX                     ; INCREMENT POINTER
677 4337 D0F7           BNE     RWTST3          ; CONTINUE UNTIL ALL RANDOM BYTES GENERATED
678 4339 A50B           LDA     TRACK           ; ESTABLISH TRACK,
679 433B 8D0840         STA     SEEKCM+2
680 433E A60C           LDX     SECTOR          ; SECTOR,
681 4340 A080           LDY     #RWTSDA/256     ; MEMORY ADDRESS,
682 4342 A50A           LDA     SIDE            ; AND DRIVE/SIDE NUMBER
683 4344 0A             ASLA
684 4345 0A             ASLA
685 4346 0509           ORA     DRIVE
686 4348 20CF43         JSR     WRITE           ; AND WRITE THE SECTOR
687 434B B079           BCS     RWERR3          ; JUMP OUT IF ANY KIND OF WRITE ERROR
688 434D E60C           INC     SECTOR          ; INCREMENT SECTOR NUMBER
689 434F A50C           LDA     SECTOR
690 4351 C91A           CMP     #26             ; TEST IF ALL SECTORS OF TRACK WRITTEN
691 4353 D0D9           BNE     RWTST2          ; GO WRITE NEXT SECTOR IF NOT
692 4355 A200           LDX     #0              ; WAIT AWHILE FOR WRITE CURRENT TO DECAY
693 4357 CA RWTST4:     DEX                     ; BEFORE SEEKING
694 4358 D0FD           BNE     RWTST4
695 435A E60B           INC     TRACK           ; INCREMENT TRACK NUMBER
696 435C A50B           LDA     TRACK
697 435E C94D           CMP     #77             ; TEST IF ALL TRACKS WRITTEN
698 4360 F00B           BEQ     RWTST5          ; GO TO READBACK PART OF TEST IF SO
```

```
699 4362 AA                TAX             ; SEEK TO THE NEW TRACK
700 4363 A509              LDA   DRIVE     ; ON DESIGNATED DRIVE
701 4365 200B41            JSR   SEEK
702 4368 B05A              BCS   RWERR4    ; JUMP OUT ON SEEK ERROR
703 436A 4C2A43            JMP   RWTST1    ; GO WRITE THE NEW TRACK
704
705 436D A901      RWTST5: LDA   #X'01     ; RESEED THE RANDOM NUMBER GENERATOR
706 436F 850F              STA   RANDNO
707 4371 8510              STA   RANDNO+1
708 4373 A200              LDX   #0        ; SEEK TO TRACK ZERO
709 4375 A509              LDA   DRIVE     ; ON DESIGNATED DRIVE
710 4377 200B41            JSR   SEEK
711 437A B048              BCS   RWERR4    ; JUMP OUT ON SEEK ERROR
712 437C A900              LDA   #0        ; SET TRACK NUMBER TO ZERO
713 437E 850B              STA   TRACK
714 4380 A900      RWTST6: LDA   #0        ; SET SECTOR NUMBER TO ZERO
715 4382 850C              STA   SECTOR
716 4384 A50B      RWTST7: LDA   TRACK     ; ESTABLISH TRACK,
717 4386 8D0840            STA   SEEKCM+2
718 4389 A60C              LDX   SECTOR    ; SECTOR,
719 438B A080              LDY   #RWTSDA/256 ; MEMORY ADDRESS,
720 438D A50A              LDA   SIDE      ; AND SIDE/DRIVE NUMBER
721 438F 0A                ASLA
722 4390 0A                ASLA
723 4391 0509              ORA   DRIVE
724 4393 20B342            JSR   READ      ; AND READ THE SECTOR INTO MEMORY
725 4396 B02A              BCS   RWERR5    ; JUMP OUT IF READ ERROR
726 4398 A200              LDX   #0        ; INITIALIZE DATA POINTER FOR RANDOM DATA
727 439A 204041    RWTST8: JSR   RAND      ; COMPARE DATA READ WITH WHAT SHOULD HAVE
728 439D DD0080            CMP   RWTSDA,X  ; BEEN WRITTEN
729 43A0 D01E              BNE   RWERR6    ; JUMP OUT IF COMPARE ERROR **SERIOUS**
730 43A2 E8                INX
731 43A3 D0F5              BNE   RWTST8    ; CONTINUE UNTIL ENTIRE SECTOR COMPARED
732 43A5 E60C              INC   SECTOR    ; INCREMENT SECTOR NUMBER
733 43A7 A50C              LDA   SECTOR
734 43A9 C91A              CMP   #26       ; TEST IF ALL SECTORS OF TRACK READ
735 43AB D0D7              BNE   RWTST7    ; GO READ NEXT SECTOR IF NOT
736 43AD E60B              INC   TRACK     ; INCREMENT TRACK NUMBER
737 43AF A50B              LDA   TRACK
738 43B1 C94D              CMP   #77       ; TEST IF ALL TRACKS CHECKED
739 43B3 F017              BEQ   RWERR0    ; GO TO MONITOR IF SO, PASSED TEST
740 43B5 AA                TAX             ; SEEK TO THE NEW TRACK
741 43B6 A509              LDA   DRIVE     ; ON DESIGNATED DRIVE
742 43B8 200B41            JSR   SEEK
743 43BB B007              BCS   RWERR4    ; JUMP OUT ON SEEK ERROR
744 43BD 4C8043            JMP   RWTST6    ; GO CHECK NEXT TRACK
745
746 43C0 E600      RWERR6: INC   ERRORC    ; 36 SET COMPARE ERROR CODE
747 43C2 E600      RWERR5: INC   ERRORC    ; 35 SET READ ERROR CODE
748 43C4 E600      RWERR4: INC   ERRORC    ; 34 SET SEEK ERROR CODE
749 43C6 E600      RWERR3: INC   ERRORC    ; 33 SET WRITE ERROR CODE
750 43C8 E600      RWERR2: INC   ERRORC    ; 32 SET RECALIBRATE ERROR CODE
751 43CA E600      RWERR1: INC   ERRORC    ; 31 SET SPECIFY ERROR CODE
752 43CC 4C2340    RWERR0: JMP   MONENT    ; 30 NO ERROR, RETURN TO MONITOR
753
```



```
754
755                ;       FLOPPY DISK WRITE SECTOR
756                ;       THIS ROUTINE WRITES ONE SECTOR FROM MEMORY ONTO DISK.
757                ;       ENTER WITH MEMORY PAGE NUMBER TO WRITE FROM IN Y (MUST BE
758                ;       IN DISK CONTROLLER RAM), SECTOR NUMBER TO WRITE TO  IN X
759                ;       WRITES ON CURRENT TRACK), AND DISK DRIVE NUMBER AND DISKETTE
760                ;       SIDE IN A.
761                ;       ***CAUTION-CURRENT CYLINDER NUMBER IS TAKEN FROM SEEKCM+2, IF
762                ;       IT DOES NOT MATCH THE ACTUAL HEAD POSITION, A WRITE ERROR WILL
763                ;       RESULT***
764                ;       RETURNS WITH CARRY CLEAR IF NO WRITE ERROR, CARRY SET IF A
765                ;       WRITE OR OTHER KIND OF ERROR.
766                ;       THE STATUS AREA IS UPDATED AFTER A WRITE, 7 BYTES ARE STORED.
767
768                ;       NOTE: USES DMA MODE BUT WAITS FOR OPERATION TO COMPLETE BEFORE
769                ;       RETURNING.
770
771 43CF 8D1B40   WRITE:  STA   WRITCM+1   ; PUT SIDE/DRIVE IN SECOND BYTE OF COMMAND
772 43D2 4A                LSRA            ; ISOLATE HEAD # AND PUT INTO H FIELD
773 43D3 4A                LSRA
774 43D4 8D1D40            STA   WRITCM+3
775 43D7 AD0840            LDA   SEEKCM+2  ; ESTABLISH CYLINDER NUMBER
776 43DA 8D1C40            STA   WRITCM+2
777 43DD 8A                TXA             ; GET SECTOR NUMBER
778 43DE 8D1E40            STA   WRITCM+4  ; SET AS SECTOR TO WRITE
779 43E1 8D2040            STA   WRITCM+6  ; SET ALSO AS LAST SECTOR TO WRITE
780 43E4 98                TYA             ; GET MEMORY PAGE ADDRESS
781 43E5 208142            JSR   DMASET    ; SET DMA ADDRESS REGISTER ACCORDINGLY
782 43E8 A900              LDA   #X'00     ; SET DMA READ MODE AND SYSTEM RAM WRITE
783 43EA 8DE89F            STA   FDCHWC    ; ALLOW
784 43ED A009              LDY   #9        ; 9 BYTES IN WRITE COMMAND
785 43EF A21A              LDX   #WRITCM-DSKCMD;RELATIVE ADDRESS OF WRITE COMMAND
786 43F1 208A40            JSR   CMDPH     ; SEND COMMAND TO DISK CONTROLLER
787 43F4 B028              BCS   WRITER    ; JUMP OUT ON ERROR
788
789 43F6 ADE89F    WRIT3:  LDA   FDCIRQ    ; WAIT UNTIL INTERRUPT REQUEST FROM FDC
790 43F9 30FB              BMI   WRIT3
791
792 43FB A200              LDX   #0        ; SET UP TO READ STATUS INTO STATUS AREA
793 43FD 20A840            JSR   RSLPH
794 4400 B01C              BCS   WRITER    ; JUMP OUT ON GROSS ERROR
795 4402 A501              LDA   DSKSTS    ; CHECK STATUS REGISTER 0
796 4404 29D8              AND   #X'D8     ; MASK OUT NON-ERROR BITS
797 4406 F008              BEQ   WRIT4     ; JUMP AHEAD IF NO OBVIOUS ERROR
798 4408 C940              CMP   #X'40     ; TEST IF ABNORMAL TERMINATION ERROR
799 440A D012              BNE   WRITER    ; TRUE ERROR IF NOT
800 440C A502              LDA   DSKSTS+1  ; IF ABNORMAL TERMINATION, TEST IF END OF
801 440E 100E              BPL   WRITER    ; CYLINDER, TRUE ERROR IF NOT, OK IF SO
802 4410 A502      WRIT4:  LDA   DSKSTS+1  ; CHECK STATUS REGISTER 1
803 4412 2935              AND   #X'35     ; MASK OUT NON-ERROR BITS
804 4414 D008              BNE   WRITER    ; GO TO ERROR IF ANY OF REMAINDER SET
805 4416 A503              LDA   DSKSTS+2  ; CHECK STATUS REGISTER 2
806 4418 2933              AND   #X'33     ; MASK OUT NON-ERROR BITS
807 441A D002              BNE   WRITER    ; GO TO ERROR IF ANY OF REMAINDER SET
808
```

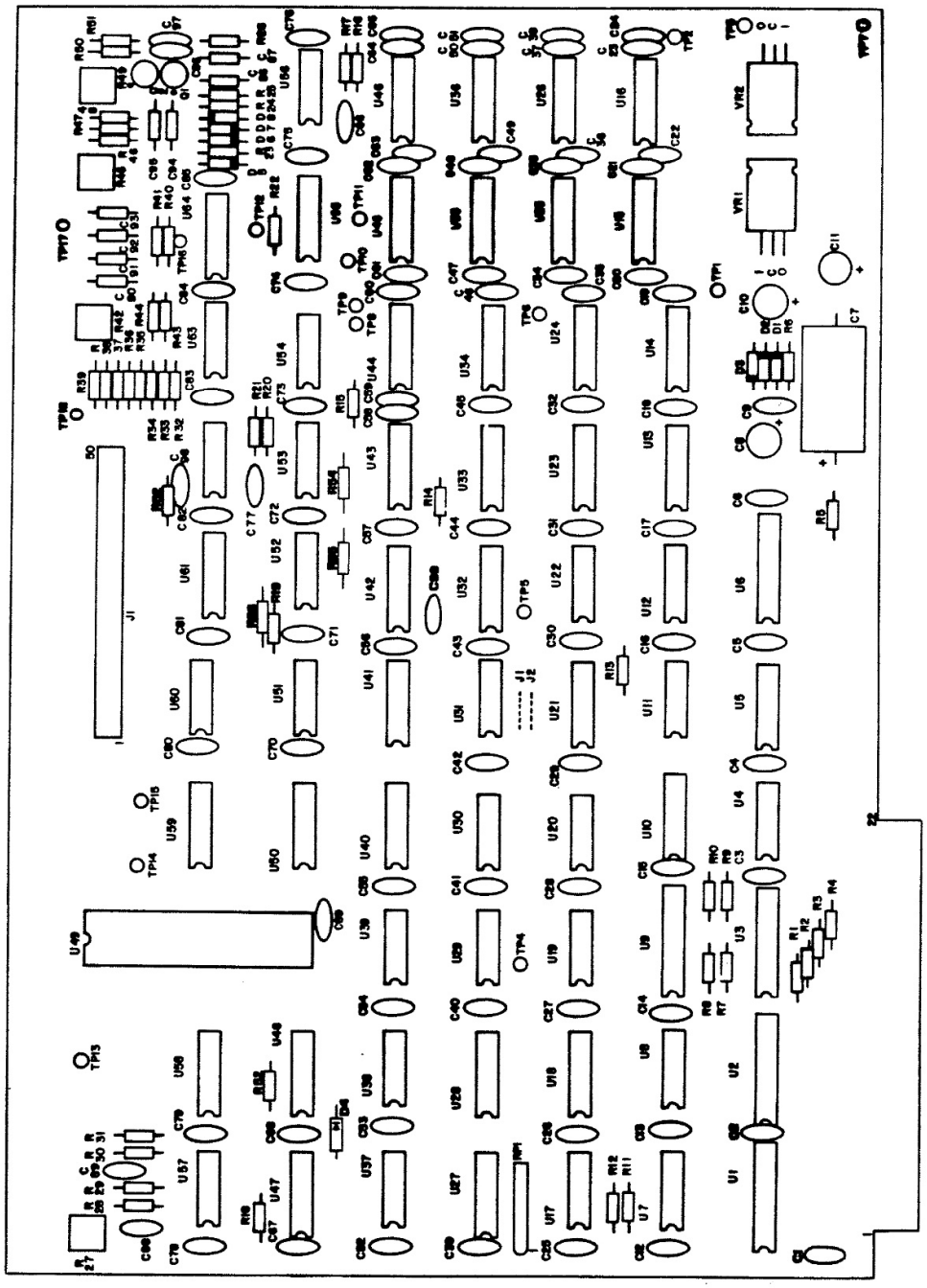
```
DSKDG  K-1013 DISK DIAGNOSTI
RANDOM DATA WRITE AND READ EXERCIZE

809 441C 18                CLC             ; CLEAR CARRY FOR NORMAL RETURN
810 441D 60                RTS
811 441E 38        WRITER: SEC             ; SET CARRY IF ANY ERROR DETECTED
812 441F 60                RTS             ; AND RETURN
813
814 0000                   .END
NO ERROR LINES
```

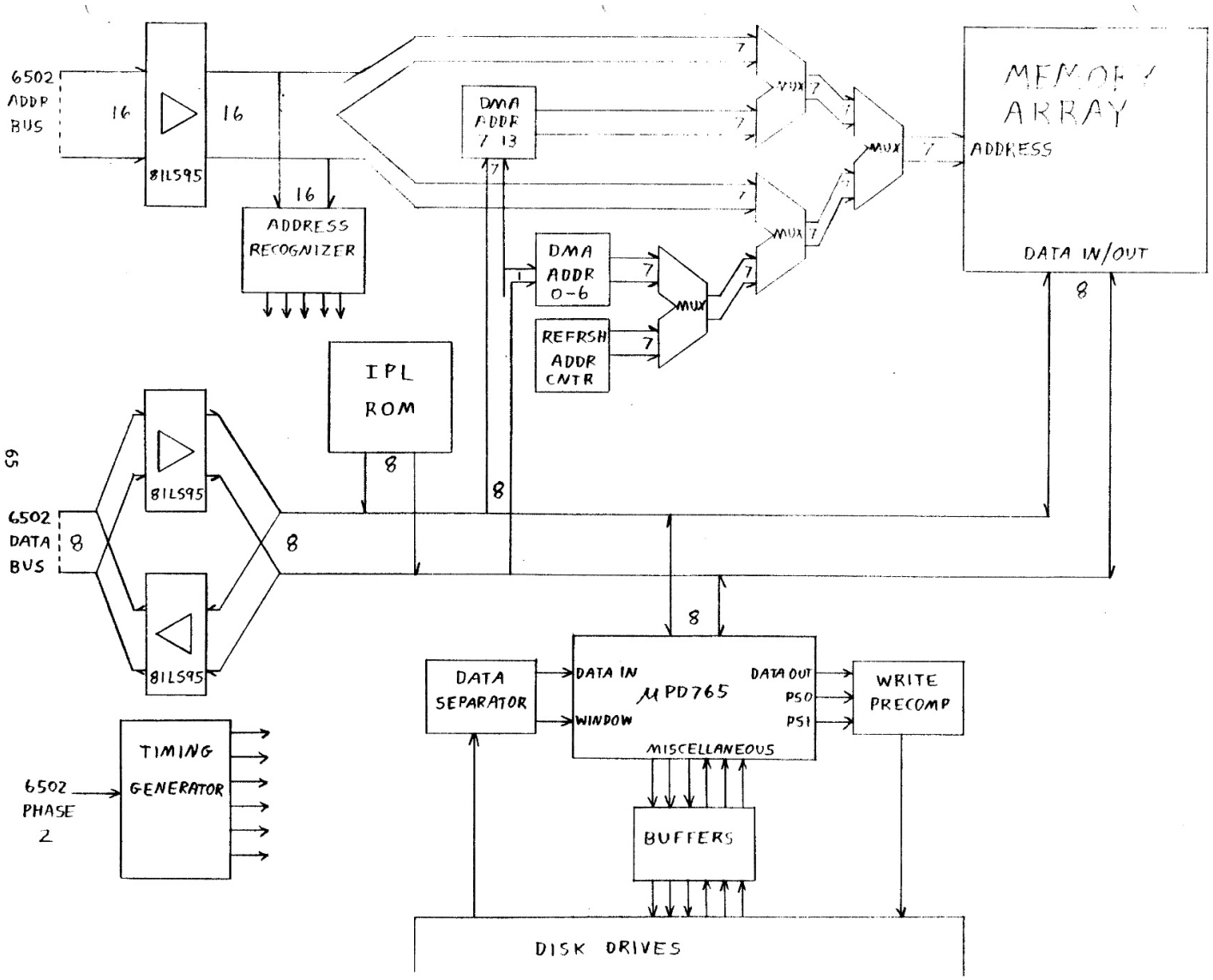
K-1013 FLOPPY DISK CONTROLLER PARTS LIST

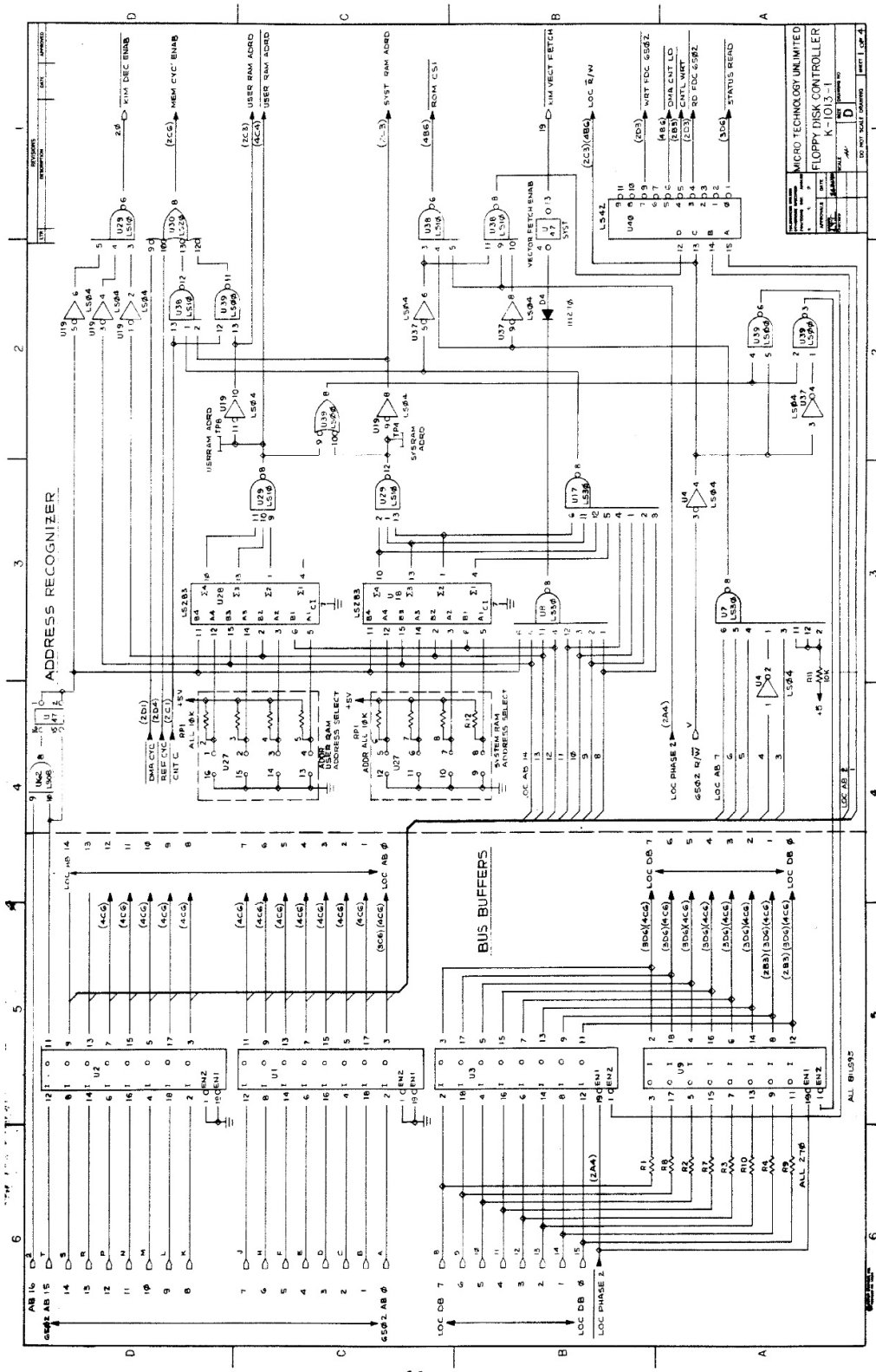
DESIGNATION	PART TYPE	QTY
U31,39,52	LOGIC, 74LS00	3
U4,19,37,51,53	LOGIC, 74LS04	5
U60	LOGIC, 7406	1
U62	LOGIC, 74LS08	1
U20,29,38	LOGIC, 74LS10	3
U56,57	LOGIC, 74LS13	2
U11,30	LOGIC, 74LS20	2
U7,8,17	LOGIC, 74LS30	3
U40	LOGIC, 74LS42	1
U63	LOGIC, 74LS51	1
U54	LOGIC, 74LS93	1
U5,10,32,42,55	LOGIC, 74LS109	5
U61	LOGIC, 74145	1
U14,24,34,44,58	LOGIC, 74LS153	5
U13,23	LOGIC, 74LS157	2
U21,33,43	LOGIC, 74LS161	3
U48	LOGIC, 74LS173	1
U64	LOGIC, 74LS221	1
U18,28	LOGIC, 74LS283	2
U50,59	LOGIC, 74LS368	2
U12,22	LOGIC, 74LS393	2
U1-3,9	LOGIC, 81LS95	4
U15,16,25,26,35, 36,45,46	MEMORY, 4116 TYPE 200ns	8
U6	MEMORY, 74S471 PROM	1
U49	LSI FD CONTROLLER UPD765	1
VR2	VOLT REG. +5V LM340T-5	1
VR1	VOLT REG. +12V LM341P-12	1
D4	DIODE, GERMANIUM 1N270	1
D2,3,5-8	DIODE, SILICON 1N4148	6
D1	DIODE, ZENER 5.1V .4W	1
Q2	TRANSISTOR, NPN PN2222	1
Q1	TRANSISTOR, PNP PN2907	1

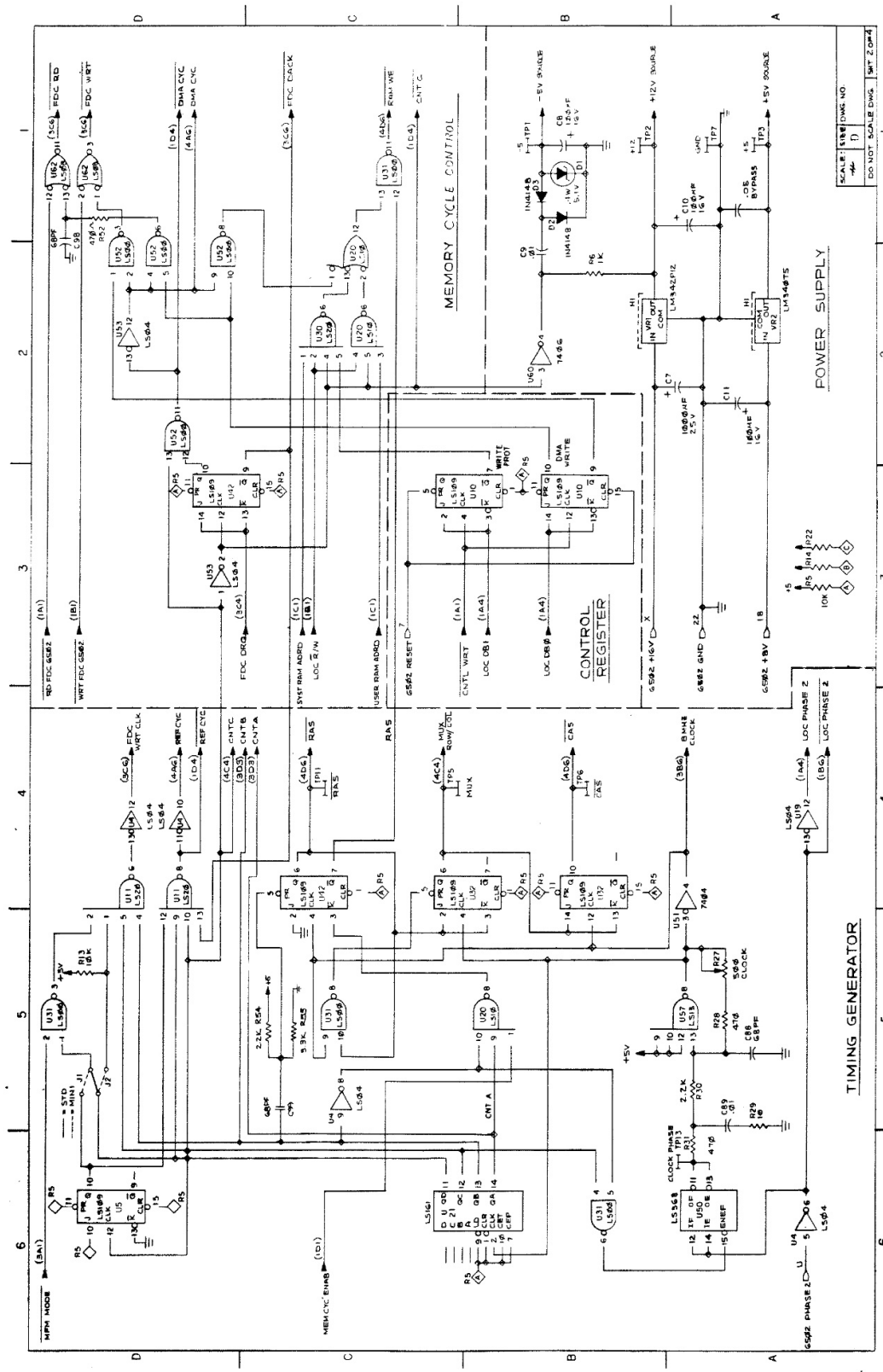
DESIGNATION	PART TYPE	QTY
	SOCKET, PC 14 PIN	24
U27,41,47	SOCKET, PC 16 PIN	35
	SOCKET, PC 20 PIN	5
	SOCKET, PC 40 PIN	1
	BOARD, PC K-1013-1 REV B	1
H1	HEATSINK 1" SQUARE AAVID 5072B	1
J1	CONNECTOR, 50 PIN HEADER	1
R29	RESISTOR, 1/4W 5% 10 OHM	1
R32,34,36,38,41	RESISTOR, 1/4W 5% 220 OHM	5
R1-4,7-10,46	RESISTOR, 1/4W 5% 270 OHM	9
R33,35,37,39,40, 52	RESISTOR, 1/4W 5% 330 OHM	6
R15,28,31,53	RESISTOR, 1/4W 5% 470 OHM	4
R6,19,26,56	RESISTOR, 1/4W 5% 1K	4
R44	RESISTOR, 1/4W 5% 1.5K	1
R16,17,20,30,54	RESISTOR, 1/4W 5% 2.2K	5
R43	RESISTOR, 1/4W 5% 3K	1
R21,55	RESISTOR, 1/4W 5% 3.3K	2
R23,24,47,48	RESISTOR, 1/4W 5% 4.7K	4
R5,11-14,18,22	RESISTOR, 1/4W 5% 10K	7
R50	RESISTOR, 1/4W 5% 51K	1
R25,51	RESISTOR, 1/4W 5% 820K	2
R27,42,45,49	TRIMPOT, 500 OHM SQUARE	4
RP1	RES. PACK 5% 10K 8 TO 5V	1
C8,10,11	CAP,ELECTROLYTIC 100UF 16V RADIAL	3
C7	CAP,ELECTROLYTIC 1000UF 25V AXIAL	1
C91,93,95	CAP, POLY, 470PF 12V	3
C87	CAP, POLY, 1000PF 12V	1
C66,88,98,99	CAP, DISK, NPO 68PF 12V	4
C58,77	CAP, DISK, Y5F 100PF 12V	2
C9,89,97	CAP, DISK, Z5U .01UF 12V	3
C1-6,12-19,25-33, 39-46,52-57,59, 60,67-76,78-85,	CAP, DISK, Z5U .047UF 12V	57
20-24,34-38, 47-51,61-65	CAP, DISK, Z5U .1UF 12V	20

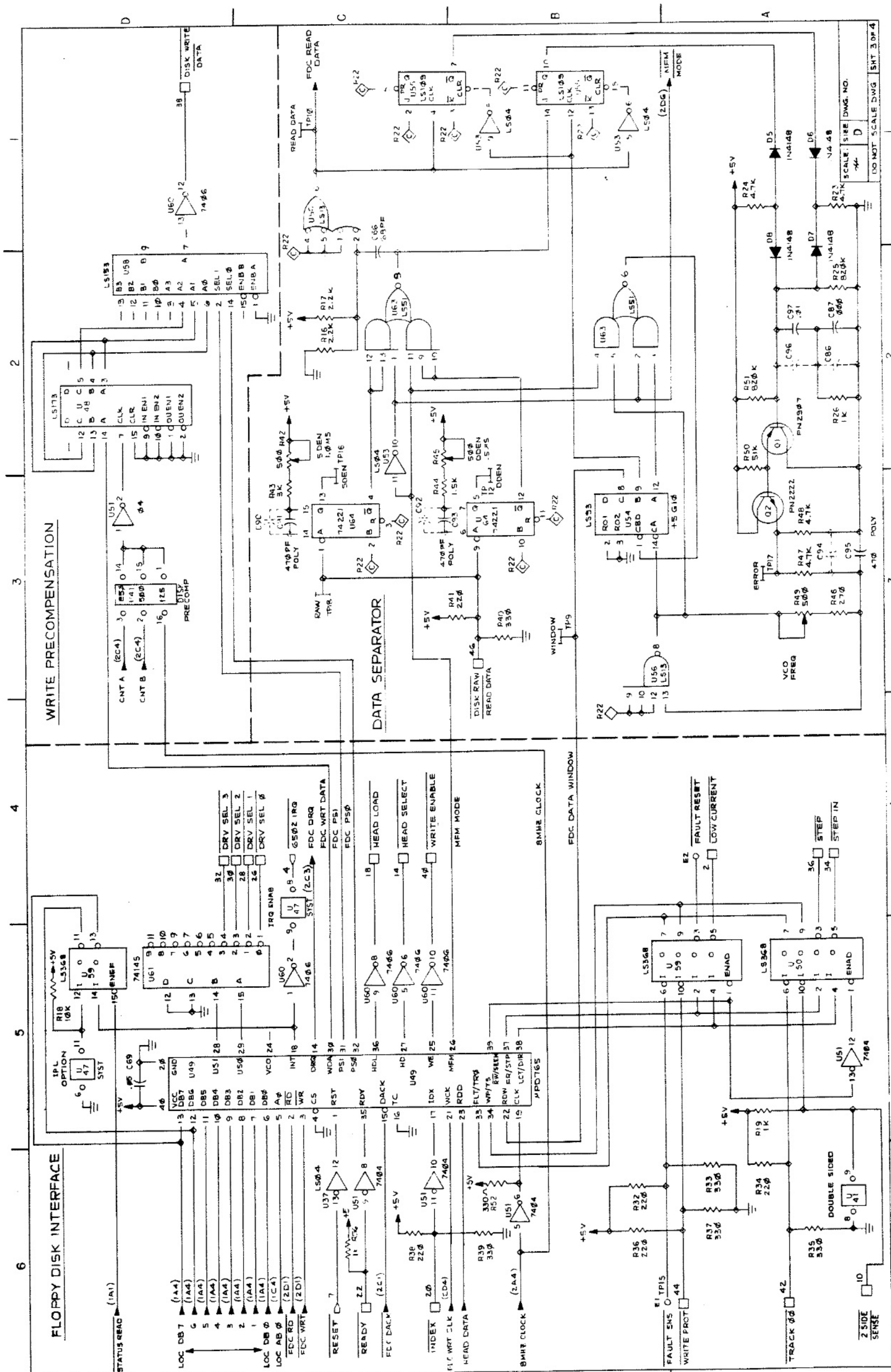
K-1013 PARTS LAYOUT

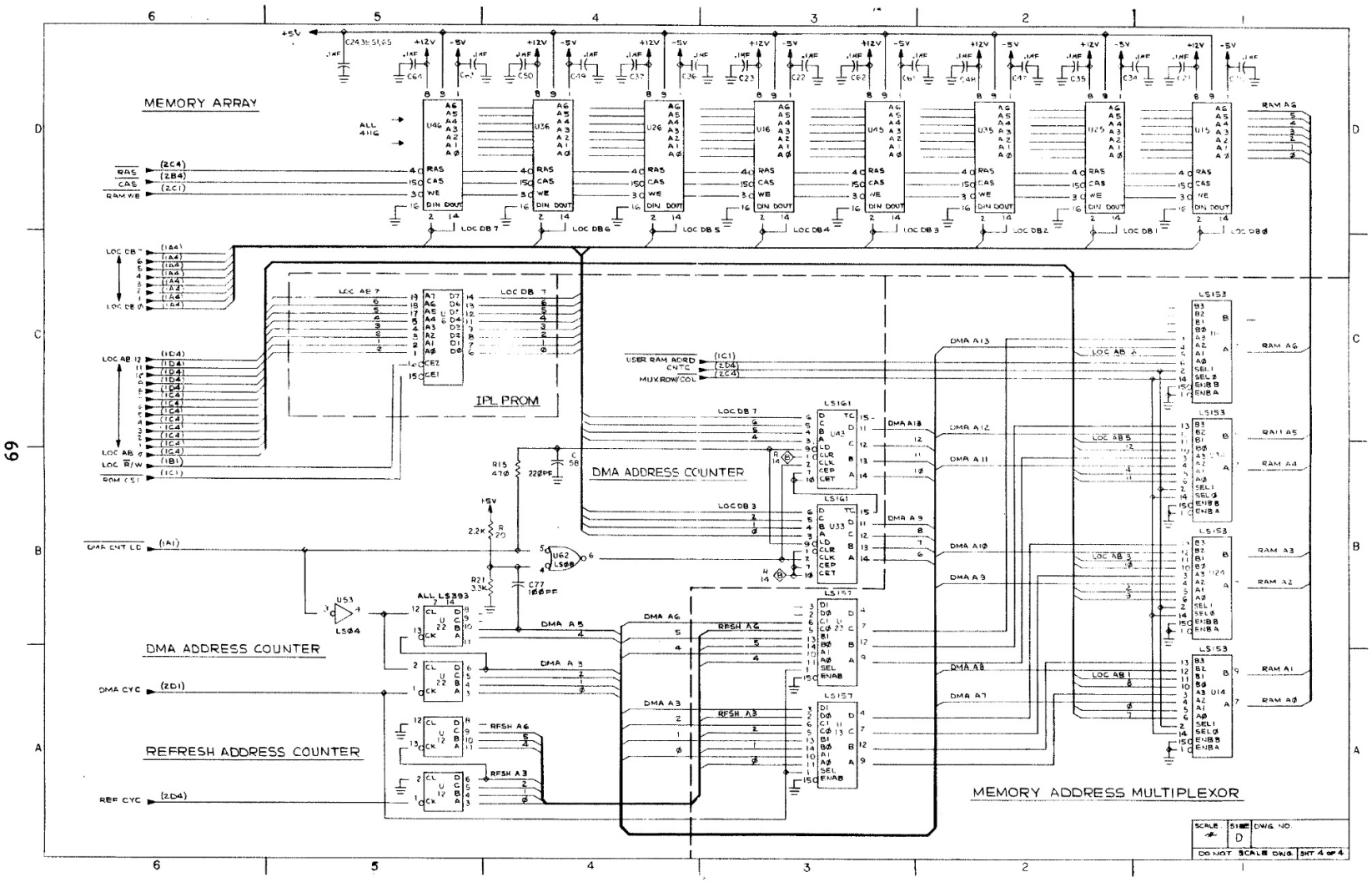



6502
ADDR
BUS
16
16
81LS95
16
ADDRESS
RECOGNIZER
DMA
ADDR
7-13
7
MUX
MUX
MUX
7
MEMORY
ARRAY
ADDRESS
DATA IN/OUT
8
DMA
ADDR
0-6
REFRSH
ADDR
CNTR
IPL
ROM
8
8
8
81LS95
81LS95
6502
DATA
BUS
8
8
8
DATA
SEPARATOR
DATA IN
WINDOW
μPD765
DATA OUT
PS0
PS1
MISCELLANEOUS
WRITE
PRECOMP
TIMING
GENERATOR
6502
PHASE
2
BUFFERS
DISK DRIVES


ADDRESS RECOGNIZER
BUS BUFFERS
MICRO TECHNOLOGY UNLIMITED
FLOPPY DISK CONTROLLER
K-1013-1


MEMORY CYCLE CONTROL
CONTROL REGISTER
POWER SUPPLY
TIMING GENERATOR
SCALE
DWG NO
DO NOT SCALE DWG.
SHT 2 OF 4


FLOPPY DISK INTERFACE
WRITE PRECOMPENSATION
DATA SEPARATOR
DISK WRITE DATA
FDC READ DATA
MFM MODE
8MHZ CLOCK
FDC DATA WINDOW
FAULT RESET
LOW CURRENT
STEP
STEP IN
HEAD LOAD
HEAD SELECT
WRITE ENABLE
STATUS READ
RESET
READY
INDEX
DOUBLE SIDED
TRACK 00
2 SIDE SENSE
WRITE PROT
DO NOT SCALE DWG
SHT 3 OF 4


MEMORY ARRAY
IPL PROM
DMA ADDRESS COUNTER
REFRESH ADDRESS COUNTER
MEMORY ADDRESS MULTIPLEXOR
DMA CNT LD
DMA CYC
REF CYC
RAS
CAS
RAM WE
LOC R/W
ROM CS1
USER RAM ADRD CNTC
MUX ROW/COL
LOC DB 7
LOC DB 6
LOC DB 5
LOC DB 4
LOC DB 3
LOC DB 2
LOC DB 1
LOC DB 0
LOC AB 7
LOC AB 12
LOC AB 0
DMA A 13
DMA A 12
DMA A 11
DMA A 10
DMA A 9
DMA A 8
DMA A 7
DMA A 6
DMA A 5
DMA A 4
DMA A 3
RFSH A6
RFSH A3
RAM A6
RAM A5
RAM A4
RAM A3
RAM A2
RAM A1
RAM A0
+5V
+12V
-5V
SCALE
SIZE
D
DWG NO.
DO NOT SCALE DWG.
SHT 4 OF 4