



**MINIFLOPPY DISK
OPERATING SYSTEM
FDOS VER. 1.0[©]**



DISK BASIC VER. 1.0[©]

Written By

ROBERT H. UITERWYK

4402 Meadowwood Way
Tampa, Florida 33624

Copyright © 1977, Southwest Technical Products Corporation

"All Rights Reserved"



SOUTHWEST TECHNICAL PRODUCTS CORPORATION
219 West Rhapsody San Antonio, Texas 78216

SWTPC MF-68 FDOS OPERATING SYSTEM

Overview

The floppy disk controller for the SWTPC MF-68 uses the 1771 floppy disk controller chip to interface up to four minifloppy disk drives to the SWTPC 6800 Computer System. The disk controller board on which this chip resides, is plugged onto interface card position #6 just like a standard serial or parallel interface card. When the Computer System is powered up, the monitor ROM prints a "*" prompt on the terminal's screen and awaits user commands, but the system has no program code in memory which allows it to communicate with or even be aware of the disk controller board and attached disk drives which are plugged onto the system.

A short, sixty byte program called the bootstrap must be loaded into the computer system either by hand or cassette or paper tape since it is not resident within the Mikbug^R monitor. This particular bootstrap must start execution at 0100, therefore program counter addresses A048 and A049 must be set to 0100 before typing the "G" to execute the bootstrap program. The bootstrap program is responsible for loading the FDOS operating system resident on the disk from memory location 2400 thru 3196. If any program or operation ever alters the contents of any of these memory locations, the system will have to be rebooted to restore the FDOS operating system. Any time machine control is turned over to a user program and the FDOS memory locations are not overwritten, you may return to FDOS by jumping to 2400 at the conclusion of your program or by resetting the program counter addresses A048 and A049 to 2400 and typing a "G" for go to user program.

The FDOS operating system uses memory locations 0000 thru 0020 for temporary storage when loading and saving programs, therefore, no user programs should use these locations for permanent data or program code storage. Since FDOS is not operating simultaneously with user programs, there is no harm in using these locations for temporary storage within user programs however.

FDOS Command Format

All commands in FDOS take the format of:

COMMAND (DRIVE NUMBER), (FILENAME), (PASSWORD)

A DRIVE NUMBER may be from 0 to 3 (provided you have 4 Drives). If you use a DRIVE NUMBER, it must be followed by a comma before the FILENAME. If no DRIVE NUMBER is used, the system presumes DRIVE NUMBER 0 and the comma should not be used. The use of a PASSWORD is optional; however, if used, it must be separated from the FILENAME with a comma. If you do not give a FILENAME, the system will prompt the user for one. (Which may be again preceded by a DRIVE NUMBER).

FILENAME and PASSWORD may be any combination of up to eight alphanumeric characters and the first of each must be alphabetic and not numeric. COMMAND's, FILENAME's and PASSWORD's should not contain lower case alphabetic characters.

Booting the FDOS System

The bootstrap program when executed causes the computer to load and execute the FDOS operating system. Since the bootstrap program is not resident in the Mikbug^R ROM, it must be loaded by hand, at least the first time. It is suggested that you make a cassette or paper tape copy of the bootstrap program so that you do not have to repeatedly enter it by hand each time the system must be booted. The bootstrap program itself resides from 0100 thru 013D while program counter locations A048 and A049 must be set to 0100 to execute the boot. Once the boot brings in FDOS, it has served its purpose and may be overwritten by subsequent user programs.

```

00010          NAM      BOOT
00020          OPT      OBJ
00040          *WRITTEN BY R. H. UITERWYK

00050      8014      DRVREG EQU      $8014
00060      8018      COMREG EQU     $8018
00070      801B      DATREG EQU     $801B
00080      801A      SECREG EQU     $801A
00100 0100          ORG      $0100
00110 0100 4F      START CLR A
00120 0101 B7 8014 STA A  DRVREG
00130 0104 CE FFFF LDX      #$FFFF
00140 0107 08      STARTO INX
00150 0108 09          DEX
00160 0109 09          DEX
00170 010A 26 FB      BNE      STARTO
00180 010C C6 0B      LDA B  #$0B      RESTORE W/LOAD
00190 010E F7 8018    STA B  COMREG
00200 0111 8D 2A      BSR      RETURN
00210 0113 F6 8018 LOOP1 LDA B  COMREG
00220 0116 C5 01      BIT B  #1
00230 0118 26 F9      BNE      LOOP1
00240 011A 7F 801A    CLR      SECREG
00250 011D 8D 1E      BSR      RETURN
00260 011F C6 9C      LDA B  #$9C
00270 0121 F7 8018    STA B  COMREG
00280 0124 8D 17      BSR      RETURN
00290 0126 CE 2400    LDX      #$2400
00300 0129 C5 02      LOOP2 BIT B  #2
00310 012B 27 06      BEQ      LOOP3
00320 012D B6 801B    LDA A  DATREG
00330 0130 A7 00      STA A  0, X
00340 0132 08          INX
00350 0133 F6 8018 LOOP3 LDA B  COMREG
00360 0136 C5 01      BIT B  #1
00370 0138 26 EF      BNE      LOOP2
00380 013A 7E 2400    JMP      $2400
00390 013D 39          RETURN RTS
00400          END

```

CATALOG

"CATALOG (DRIVE)" or "CAT (DRIVE)" - Typing either of these commands causes the system to list all files stored on the diskette. Files are listed three to a line (See FILES and PRINT).

EXAMPLES: CATALOG ; CAT 1

FILES

"FILES (DRIVE)" - This command will list diskette files along with directroy information. The data output is:

TRACK	SECTOR	#OF SECTORS USED	FILE TYPE	STARTING ADDRESS	ENDING ADDRESS	PROGRAM START
-------	--------	------------------------	--------------	---------------------	-------------------	------------------

FILE TYPE NUMBERS ARE:

00	BLANK FILE
11	SYSTEM FILE
22	OBJECT PROGRAM FILE
55	BASIC PROGRAM FILE
77	CORES SOURCE FILE
99	BASIC DATA FILE

EXAMPLES: FILES ; FILES 1

PRINT

"PRINT" - This command causes the command following the PRINT command (which presumably would be CAT or FILES) to be output to a SWTPC PR-40 Printer on Port #7 via an MP-L parallel interface.

EXECUTING SYSTEM FILE PROGRAMS SUCH AS BASIC AND CORES

System File programs such as "BASIC" and "CORES" are loaded and executed simply by typing the file name. All commands for loading, saving and executing the source files created and manipulated by these system file programs are contained within the system file programs themselves rather than the FDOS operating system. Details on the operation of these non-FDOS commands are contained in the accompanying user's guides for the BASIC and CORES system file programs.

EXAMPLES: BASIC ; CORES

SAVE

Any memory resident machine language program may be stored to the disk as an object program file by typing the word "SAVE" or the word "SAVE" followed by the file name and password (if applicable). The terminal responds by asking for the starting address, ending address, and beginning execution address of the particular program which you wish to save. The operating system then allocates 25% more space than is necessary to store your program to allow for future expansion. If the additional space is not desired, then the space may be allocated using the "CREATE" command which is described below. File names may be followed by a password for file protection, if desired. In order to protect the file, type the FILE NAME ", " PASSWORD. Passwords do not appear in the catalog and are inaccessible. They must be remembered by the user in order to regain access to the file.

EXAMPLES: SAVE JUNKNINE ; SAVE TESTTWO,SPECIAL ; SAVE2,TESTONE

LOAD

Any disk resident object program file may be restored to a memory resident machine language program by typing the word "LOAD" or the word "LOAD" followed by the file name and password (if applicable). If a password was used when the program was "SAVE"ed, the file name must be followed by a comma and the password. If an invalid password is used on a protected file, then an error message is presented. The "LOAD" command simply LOADS program code. The program is loaded into memory but is not executed. Program counter addresses A048 and A049 are set to the starting address of the program. The program may be executed by exiting the FDOS operating system using the "EXIT" command and typing a "G" for "Go to user program".

EXAMPLES: LOAD 2,TESTONE ; LOAD JUNKNINE ; LOAD TESTTWO,SPECIAL

RUN

The RUN command loads and executes any disk resident object program file. It works just like the "LOAD" command except that you do not have to "EXIT" FDOS and type a "G" to execute the program. This is done automatically for you by FDOS. It is initiated by typing the word "RUN" or the word "RUN" followed by the file name and password (if applicable).

Upon completion of your program, you may return to FDOS by resetting the program counter address A048 and A049 to 2400 and typing a "G" only if your program has not overwritten any memory addresses above 2400₁₆. If your program has overwritten memory address above 2400₁₆ then it will be necessary to

re-bootstrap the system.

EXAMPLES: RUN LUNAR ; RUN1,ANIMALS

CREATE

"CREATE" - This command is used to allocate a fixed number of sectors of file space to be used later when saving a program. After typing "CREATE", the system will ask for a file name and the number of sectors to be allocated. This file space will be allocated in the catalog and will be reserved for saving a program having the same name. Each sector is 256 bytes long.

EXAMPLES: CREATE JUNKNINE, CREATE TESTFIVE,ROBERT

INIT

"INIT" - Typing "INIT" causes the system to initialize a new diskette. The system will respond with the question "ARE YOU SURE?" to prevent accidental initialization of a disk containing programs. Do not initialize a diskette unless you wish to erase it completely since your catalogs and programs will be gone. The initialized function stores the disk driver routines and FDOS routines on Tracks 00 and 01 of the disk. FDOS Directory Entries are stored on Track 02 of the disk. The initialize routine MUST be performed before utilizing a diskette.

EXAMPLES: INIT ; INIT 2

COPY

"COPY" - This command copies the contents of the diskette in DRIVE 0 onto the diskette in DRIVE 1. Any old contents on the destination diskette are destroyed. You must "INIT" a diskette before you copy onto it if it has never been "INIT" ialized.

DELETE or PURGE

"DELETE" or "PURGE" - These commands delete a file name from the diskette catalog, but do not pack the diskette. FDOS will not allow system files to be deleted.

EXAMPLES: DELETE JUNK ; PURGE1,TESTFILE

PACK

"PACK" - This command packs a diskette that has deleted files on it. This command can take a considerable amount of time and thus should be used sparingly. Also, as this command has to physically move files, it is the one command that could "BOMB" your diskette. Therefore, a wise precaution would be to make a back-up diskette using the "COPY" command before you "PACK" the diskette. The pack command can only be used on drive 0.

EXAMPLE: PACK

RENAME

"RENAME" - This command allows file names to be changed. FDOS will ask for the new FILENAME. (This command can also be used to rename an unpassworded file to a passworded file and vice versa.)

EXAMPLES: RENAME 1,TESTFILE ; RENAME JUNK

HOME

"HOME" - This command causes FDOS to reset the head on the designated disk to the Track "0" position. It is used mainly as a diagnostic tool.

EXAMPLES: HOME ; HOME 3

EXIT

"EXIT" - This command returns the user to the ROM monitor system.

TEST

"TEST" - This command reads all tracks and sectors on a diskette and verifies cyclic redundancy check numbers. The diskette is not written upon by this command. The diskette must be at least "INIT" before using this command. You will get a list of all bad tracks and sectors.

EXAMPLES: TEST , TEST 3

CUSTOMIZING DISK I/O

For those wishing to do advanced programming, a source listing of the I/O drivers has been supplied. By using these drivers, information can be directly removed from or stored to a disk. For example, if you wanted to read in 100 (hex) bytes from track 10, sector 2 of drive 0 and store it from 1000 to 1100 the following sequence would be used:

- 1) Store the track desired in the temporary storage location TRACK. 0
- 2) Store the sector desired in the temporary storage location SECTOR. 1
- 3) Store the drive number in NDRIVE. 0
- 4) Store the ending memory location in EMEMH. 67
- 5) Load the index register with the starting memory location.
- 6) JSR READ. This will read in the information and store it in memory. Any subsequent reads can be made by using the subroutine READ0. This is the same routine as READ except that it does not home the head and does not jump to READY.

→ To write information on the diskette, a similar procedure is followed using the same temporary storage locations for TRACK, SECTOR, DRIVE #, and ENDING ADDRESS, and by loading the index registers with the beginning address. The subroutine WRITE should be used to store information, with WRITE0 being used after WRITE is used the first time. When writing to a diskette, one complete sector must be written. If starting and ending memory locations specify data that will not fill the sector, the remainder of the sector will be filled with 0's.

Each diskette is formatted in IBM 256 compatible format with shorter inter-record gaps as recommended by the drive manufacturer. This allows 34 tracks, 10 sectors/track, 256 bytes/sector.

When writing directly to a diskette, be very careful. FDOS is contained on tracks 0 and 1 and track 2 contains the disk catalog. When directly adding files the disk catalog must be updated each time, and this is not an easy, straightforward process. Each entry in the catalog is 20 (hex) bytes long and is organized as follows:

FILE NAME	(8 bytes)
PASSWORD	(8 bytes)
STARTING TRACK	(1 byte)
STARTING SECTOR	(1 byte)
NUMBER OF SECTORS	(2 bytes)
FILE TYPE	(1 byte)
STARTING ADD.	(2 bytes)
ENDING ADD.	(2 bytes)
PROGRAM START	(program counter) (2 bytes)
HIGH LINE #	(2 bytes) Used for BASIC and CO-RES files
SPARES	(3 bytes)

After the last catalog entry, an FF is stored in what would be the NAME of the next entry. In the STARTING TRACK location is stored the next available track, the STARTING SECTOR contains the next available sector and the NUMBER of SECTORS location contains the number of remaining sectors.

DISK NOTES

FDOS uses the I/O select line of I/O port #5 as the drive MOTOR ON signal. In some cases, when addressing a printer on another I/O slot, the drive motors will activate. This doesn't hurt anything, but may be an annoyance in certain cases. If this causes a problem, jumper a short wire from IC5 pin 1 to IC3 pin 6 on the computer's MP-B mother board.

For those of you who are interested in the workings of the 1771 disk controller used in the MF-68, a lengthy discussion can be found in the October, November and December 1976 issues of Interface Age Magazine.

Any time a DISK ERROR message is received, two four digit hex numbers will be displayed. The error message format is as follows:

TRACK	SECTOR	ERROR	# OF ATTEMPTS MADE
XX	XX	XX	XX

Errors: 11 Seek error
08 CRC error (bad data)
80 No drive power

Useful Memory Locations

2400		Start of boot drivers
2600	- 2FFF	FDOS
3000	- 31FF	Temporary FDOS storage
3077	- 3176	256 byte buffer for catalog reads
0000	- 001F	Temporary FDOS storage

00010
00020

NAM SWTFID
*****VERSION 1 0 *****

00030

*WRITTEN BY R. H. UITERWYK

00040

OPT NOG

00050

OPT OBU

00070

2600 DOS EQU \$2600

00080

8014 DRVREG EQU \$8014

00090

8018 COMREG EQU \$8018

00100

8019 TRKREG EQU \$8019

00110

801A SECREG EQU \$801A

00120

801B DATREG EQU \$801B

801C
801D
801E
801F

00150

0000 ORG \$0000

00160

0000 0001 / TRACK RMB 1

00170

0001 0001 / SECTOR RMB 1

00180

0002 0001 STATUS RMB 1

00190

0003 0001 ERRCNT RMB 1

00200

0004 0001 / BMEMH RMB 1

00210

0005 0001 / BMEML RMB 1

00220

0006 0001 / EMEMH RMB 1

00230

0007 0001 / EMEML RMB 1

00240

0008 0001 / NDSECT RMB 1

00250

0009 0002 SAVEX RMB 2

00260

000B 0002 / PROGX RMB 2

00270

000D 0001 NDRIVE RMB 1

00280

000E 0004 ATRACK RMB 4

00290

0012 0002 PRELEX RMB 2

00310

2400 ORG \$2400

00320

2400 BD 240C START JSR BOOT

00330

2403 DE 0B RESTRT LDX PROGX

00340

2405 8E A049 LDB #\$A049

00350

2408 AD 00 JSR O. X

00360

240A 20 F7 BRA RESTRT

00370

240C CE 240C BOOT LDX #BOOT

00380

240F DF 0B STX PROGX

00390

2411 4F CLR A

00400

2412 97 0D STA A NDRIVE

00410

2414 97 00 STA A TRACK

00420

2416 86 02 LDA A #2

00430

2418 97 01 STA A SECTOR

00440

241A CE 2FFF LDX ##2FFF

00450

241D DF 06 STX EMEMH

00460

241F CE 2600 LDX #DOS

00470

2422 8D 07 BSR READ

00480

2424 CE 2600 LDX #DOS

00490

2427 DF 0B STX PROGX

00500

2429 8E 00 JMP O. X

00520

242B BD 2523 READ JSR PRELIM

00530

242E BD 250E READ0 JSR SEEK

00540

2431 DF 04 READ2 STX BMEMH

00550

2433 7F 0003 CLR ERRCNT

00560	2436	DE	04	READ3	LDX	BMEMH	
00370	2438	D6	01		LDA B	SECTOR	
00580	243A	F7	801A		STA B	SECUREG	
00590	243D	BD	2489		JSR	RETURN	
00600	2440	C6	9C		LDA B	##9C	MULTIPLE READ & LOAD
00610	2442	F7	8018		STA B	COMREG	IBM FORM
00620	2445	BD	2489		JSR	RETURN	DELAY 14 CYCLES
00630	2448	BD	2489		JSR	RETURN	
00640	244B	BD	2489		JSR	RETURN	
00650	244E	F6	8018	READ4	LDA B	COMREG	
00660	2451	C5	01		BIT B	#1	BUSY FLAG
00670	2453	27	10		BEQ	READ6	
00680	2455	C5	02	READ5	BIT B	#2	DRQ FLAG
00690	2457	27	F5		BEQ	READ4	
00700	2459	B6	801B		LDA A	DATREG	
00710	245C	A7	00		STA A	0. X	
00720	245E	9C	06		CPX	EMEMH	
00730	2460	27	0C		BEQ	READ7	
00740	2462	08			INX		
00750	2463	20	E9		BRA	READ4	
00760	2465	8D	0F	READ6	BSR	CHKERR	
00770	2467	26	CD		BNE	READ3	
00780	2469	BD	24F8		JSR	INCSER	
00790	246C	20	C0		BRA	READ0	
00800	246E	8D	06	READ7	BSR	CHKERR	
00810	2470	26	C4		BNE	READ3	
00820	2472	BD	24F8		JSR	INCSER	
00830	2475	39			RTS		
00850	2476	F6	8018	CHKERR	LDA B	COMREG	
00860	2479	D7	02		STA B	STATUS	
00870	247B	86	D0		LDA A	##D0	FORCE INTERRUPT
00880	247D	B7	8018		STA A	COMREG	
00890	2480	BD	25D7		JSR	DONE	
00900	2483	D6	02		LDA B	STATUS	
00910	2485	C4	1C		AND B	##1C	
00920	2487	26	01		BNE	CHKER1	
00930	2489	39		RETURN	RTS		
00940	248A	C4	10	CHKER1	AND B	##10	
00950	248C	27	09		BEQ	CHKER3	
00960	248E	F6	801A		LDA B	SECUREG	
00970	2491	C1	0A		CMP B	#10	
00980	2493	26	02		BNE	CHKER3	
00990	2495	5F			CLR B		
01000	2496	39			RTS		
01010	2497	D6	03	CHKER3	LDA B	ERRCNT	
01020	2499	5C			INC B		
01030	249A	D7	03		STA B	ERRCNT	
01040	249C	C1	06		CMP B	#6	
01050	249E	26	E9		BNE	RETURN	
01060	24A0	F6	801A		LDA B	SECUREG	
01070	24A3	D7	01		STA B	SECTOR	
01080	24A5	CE	0000		LDX	#TRACK	
01090	24A8	8D	4B		BSR	OUT4HS	

01100	24AA	8D	49		BSR	OUT4HS	
01110	24AC	CE	24D4		LDX	#CRLF	
01120	24AF	8D	41		BSR	PDATA	
01130	24B1	CE	24BE		LDX	#CHKMSG	
01140	24B4	8D	3C	ERROR	BSR	PDATA	
01150	24B6	CE	24C7		LDX	#ERRMSG	
01160	24B9	8D	37		BSR	PDATA	
01170	24BB	7E	2403		JMP	RESTRT	
01190	24BE	0D		CHKMSG	FCB	\$0D, \$0A, \$15	
01200	24C1	43			FCC	/CHECK/	
01210	24C6	04			FCB	4	
01220	24C7	20		ERRMSG	FCC	/ - DISK ERROR/	
01230	24D4	0D		CRLF	FCB	\$0D, \$0A, \$15, 4	
01240	24D8	0D		RDYMSG	FCB	\$0D, \$0A, \$15	
01250	24DB	4E			FCC	/NOT READY/	
01260	24E4	04			FCB	4	
01270	24E5	0D		PROMSG	FCB	\$0D, \$0A, \$15	
01280	24E8	50			FCC	/PROTECTED/	
01290	24F1	04			FCB	4	
01300	24F2	7E	E07E	PDATA	JMP	\$E07E	
01310	24F5	7E	E0C8	OUT4HS	JMP	\$E0C8	
01330	24F8	37		INCSEC	PSH	B	
01340	24F9	F6	801A		LDA	B	SECRET
01350	24FC	C1	0A		CMF	B	#10
01360	24FE	26	07		BNE		INCSE2
01370	2500	7C	0000		INC		TRACK
01380	2503	BD	250E		JSR		SEEK
01390	2506	5F			CLR	B	
01400	2507	D7	01	INCSE2	STA	B	SECTOR
01410	2509	F7	801A		STA	B	SECRET
01420	250C	33			PUL	B	
01430	250D	39			RTS		
01450	250E	D6	00	SEEK	LDA	B	TRACK
01460	2510	F7	801B		STA	B	DATREG
01470	2513	BD	2489		JSR		RETURN
01480	2516	C6	1B		LDA	B	#\$1E LOAD & SEEK
01490	2518	F7	8018		STA	B	COMREG
01510	251B	BD	25D7		JSR		DONE
01520	251E	C5	10		BIT	B	#\$10
01530	2520	26	EC		BNE		SEEK
01540	2522	39			RTS		
01560	2523	36		PRELIM	PSH	A	
01570	2524	96	0D		LDA	A	NDRIVE
01580	2526	BD	25AC		JSR		DRIVE
01590	2529	32			PUL	A	
01600	252A	DF	12		STX		PRELEX
01610	252C	F6	8018		LDA	B	COMREG
01620	252F	CE	4FFF		LDX		#\$4FFF

01630	2532	08		PREL25	INX		
01640	2533	09			DEX		
01650	2534	09			DEX		
01660	2535	26	FB		BNE	PREL25	
01670	2537	C6	0B		LDA B	##0B	LOAD + RESTORE
01680	2539	F7	8018		STA B	COMREG	
01690	253C	BD	25D7		JSR	DONE	
01700	253F	CE	4000		LDX	##4000	
01710	2542	F6	8018	PREL35	LDA B	COMREG	
01720	2545	09			DEX		
01730	2546	27	11		BEQ	PREL4	
01740	2548	C5	02		BIT B	##02	
01750	254A	26	F6		BNE	PREL35	
01760	254C	F6	8018	PREL36	LDA B	COMREG	
01770	254F	09			DEX		
01780	2550	27	07		BEQ	PREL4	
01790	2552	C5	02		BIT B	##02	
01800	2554	27	F6		BEQ	PREL36	
01810	2556	DE	12		LDX	FRELEX	
01820	2558	39			RTS		
01830	2559	CE	24D8	PREL4	LDX	#RDYMSG	
01840	255C	7E	24B4	PREL45	JMF	ERROR	
01880	255F	8D	C2	WRITE	BSR	PRELIM	
01890	2561	8D	AB	WRITE0	BSR	SEEK	
01900	2563	7F	0003	WRITE1	CLR	ERRCNT	
01910	2566	DF	04		STX	BMEMH	
01920	2568	DE	04	WRITE2	LDX	BMEMH	
01930	256A	D6	01		LDA B	SECTOR	
01940	256C	F7	801A		STA B	SECREG	
01950	256F	BD	2489		JSR	RETURN	
01960	2572	C6	BC		LDA B	##BC	MULTIPLE WRITE w/LOAD
01970	2574	F7	8018		STA B	COMREG	IBM FORM
01980	2577	BD	2489		JSR	RETURN	
01990	257A	BD	2489		JSR	RETURN	
02000	257D	BD	2489		JSR	RETURN	
02010	2580	F6	8018	WRITE3	LDA B	COMREG	
02020	2583	C5	01		BIT B	#1	BUSYFLAG
02030	2585	27	10		BEQ	WRITE6	
02040	2587	C5	02	WRITE4	BIT B	#2	DRQ FLAG
02050	2589	27	F5		BEQ	WRITE3	
02060	258B	A6	00		LDA A	O, X	
02070	258D	B7	801B		STA A	DATREG	
02080	2590	9C	06		CFX	EMEMH	
02090	2592	27	11		BEQ	WRITE7	
02100	2594	08			INX		
02110	2595	20	E9		BRA	WRITE3	
02120	2597	C5	40	WRITE6	BIT B	##40	
02130	2599	27	05		BEQ	*+7	
02140	259B	CE	24E5		LDX	#PROMSG	
02150	259E	20	BC		BRA	PREL45	
02160	25A0	BD	24F8		JSR	INC3EC	
02170	25A3	20	BE		BRA	WRITE1	

02180	25A5	BD	25D7	WRITE7	JSR		DONE
02190	25A6	BD	24F8		JSR		INCSEC
02200	25A8	39			RTS		
02220	25AC	84	03	DRIVE	AND	A	#803
02230	25AE	36			PSH	A	
02240	25AF	DF	12		STY		FRELEX
02250	25B1	CE	000D		LDX		#ATRACK-1
02260	25B4	4C			INC	A	
02270	25B5	16			TAB		
02280	25B6	96	0D		LDA	A	NDRIVE
02290	25B8	4C			INC	A	
02300	25B9	08		DRIVE0	INX		
02310	25BA	4A			DEC	A	
02320	25BB	26	FC		BNE		DRIVE0
02330	25BD	B6	8019		LDA	A	TRNREG
02340	25C0	A7	00		STA	A	00X
02350	25C2	CE	000D		LDX		#ATRACK-1
02360	25C5	08		DRIVE1	INX		
02370	25C6	5A			DEC	B	
02380	25C7	26	FC		BNE		DRIVE1
02390	25C9	A6	00		LDA	A	00
02400	25CB	B7	8019		STA	A	TRNREG
02410	25CE	32			PUL	A	
02420	25CF	97	0D		STA	A	NDRIVE
02430	25D1	B7	8014		STA	A	DRVREG
02440	25D4	DE	12		LDX		FRELEX
02450	25D6	39			RTS		
02470	25D7	BD	2489	DONE	JSR		RETURN
02480	25DA	BD	2489		JSR		RETURN
02490	25DD	F6	8018		LDA	B	COMREG
02500	25E0	C5	01		BIT	B	#1
02510	25E2	26	F3		BNE		DONE
02520	25E4	39			RTS		
02540					END		

IN CASE OF PROBLEMS

If your MF-68 fails to operate properly we suggest that you first go back and double check all parts. Be sure that they are turned as shown on the drawings and that they are the correct part number. The majority of problems turn out to be incorrect assembly. Using the printed pattern as a guide look over the board for solder bridges. Accidental solder bridges are the second most common problem in kits that are returned for repair. Be sure that all jumpers called for are in place and that all connections have been soldered.

If you suspect that the "Shugart" SA-400 drive unit itself is not working properly, remove the drive and return it to us for testing. Do not attempt to adjust, or repair the drive unit. Special equipment and tools are required and considerable damage can be done by attempting to work on these units without proper training.

REPAIR SERVICE

If you have a problem that you cannot solve, the kit may be returned for factory service. Please return the entire* kit (Chassis, case, PC boards, cable, drives, etc.) Be sure to include the supplied diskette containing FDOS.

*Remove the power transformer. This reduces postal costs and damage. Selected individual boards may be returned but in most cases it is best to send the entire unit. Do not remove the boards from the drives themselves and send them in. Repairs are performed for a flat labor charge per board plus parts and postage.

CIRCUIT

Labor Charge

Controller board and cable	\$16.00
Power supply	\$10.00
Disk drives	Depends on individual drive

If we find that the board, drive or complete unit is functional as received and does not require service, the Checkout Charge is \$10.00.

A confirmation sheet will be sent upon receipt of the kit. Please do not ask for an estimate or a detailed report on exactly what was done in repairing your unit as we cannot provide this service.

It is not necessary to enclose any funds with the kit, you will be billed for authorized repairs.

ADDITIONAL DISKETTES

Additional diskettes are available at \$5.50 ea. Part # FD-M

MF-68 Disk Instructions Addendum

When connecting the ribbon cables to the drives the instructions state to connect the cables with pin 1 of the connectors toward the top

- Backhand disk target cartons with cables may have the two card edge connectors. Installed upside down may have no numbers in a box. If you cable has black connectors with a white painted dot on one end the end with the dot is the pin 1 end. All cables should be installed as in the following
- ~~Factor~~ all relevant the cable through the end hole in the rear of the chassis the only natural way for the connectors to fit will be the right way, as shown. Be very careful to install the cable correctly - incorrect
- ~~Ship~~ ~~to~~ ~~add~~ ~~an~~ ~~UPS~~ ~~use~~ ~~damage~~ ~~to~~ ~~the~~ ~~controller~~ ~~and~~ ~~the~~ ~~drives~~ ~~repairs~~ sent by bus.

• Ship to:

Southwest Technical Products Corp.
 Repair Department - Digital Group
 Install 12 1/2 in. by 5 1/2 in. by 5 1/2 in.
 City Antonio, Texas 78216

TOP
VIEW

