

```

+++++
+
+   DOCUMENTATION FOR   +
+
+   NEWDOS+             +
+
+ Disk System for TRS-80 +
+
+++++

```

NEWDOS AND BASIC MODS	PAGES 2-24
DOS NOMENCLATURE AND STRUCTURE	5-9
REDUCED-SIZE NEWDOS	3
COMPATIBLE W/TRSDOS FILES & VICE VERSA	3
BASIC ONE-STEP ENTRY	18-19
BASIC LIST, EDIT, DELETE ABBREVIATIONS	22
BASIC CMD'XX' FOR DOS ROUTINES	16-17
BASIC SCROLLING COMMANDS	16
BASIC OPEN'E': ADD TO SEQUENTIAL FILES	14
BASIC ENTRY CORRECTION OF SHIFT ERRORS	13-14
COPY: ON ONE DRIVE, NEW BACKUP, ETC.	19
DISKDUMP: VIDEO-OR-PRINTER OPTION	22
FORMAT IMPROVEMENTS	17-18
KEYBOARD DEBOUNCE	13
REF: VARIABLES CROSS-REFERENCING OPTIONS	21
RENUM: BASIC LINES RENUMBERING OPTIONS	21-22
SCREEN-PRINT (JKL OPTION) ON LINE-PRINTER	13
DIRCHECK: TEST AND LIST DISK DIRECTORY	9
DISASSEM: MACHINE-CODE DISASSEMBLER (Z-80)	25-27
EDTASM IMPROVEMENTS, TAPE & DISK I/O	22-23
LEVEL I IN LEVEL II (BASIC1 & LVIDSKSL)	23-24
LMOFFSET: MACHINE CODE TAPE/DISK TRANSFERS	28-30
SUPERZAP: DISPLAY/PRINT/MODIFY MEM & DISKS	30-32

USE OF THE FOLLOWING PROGRAMS WILL REQUIRE PURCHASE OF RADIO SHACK'S TRSDOS DISK OPERATING SYSTEM, V.2.1 OR LATER. THE EDTASM MODIFICATIONS IN NEWDOS+ WILL REQUIRE PURCHASE OF THE RADIO SHACK EDITOR/ASSEMBLER PACKAGE.

DO NOT COPY THIS SOFTWARE FOR OTHERS - RECOMMEND ITS PURCHASE! PIRATED SOFTWARE REDUCES THE RETURN TO THOSE WHO WRITE AND MARKET THE BEST PROGRAMS, AND ULTIMATELY DEPRIVES YOU OF FURTHER WORKS FROM YOUR FAVORITE AUTHORS.

00100 ; NEWDOS+ INFORMATION - 03/15/79
00110 ; HEREIN ARE COMMENTS ON THE CHANGES MADE TO THE TRS-80 DOS SYSTEM BY
00120 ; APPARAT, INC., 6000 E. EVANS AVE., DENVER CO 80222 (303/758-7275)
00130 ; MINOR EDITING AND BROCHURE PREPARATION BY
00140 ; MILLER MICROCOMPUTER SERVICES, 61 LAKE SHORE RD., NATICK MA 01760 (617/653-6136)
00150 ;
00160 ;
00170 ; *****
00180 ; WITH THESE CHANGES, APPARAT DOES NOT PROVIDE COPIES OR EXTRACTS OF THE ORIGINAL
00190 ; DOCUMENTATION FOR BASIC, FOR DOS OR FOR THE EDITOR-ASSEMBLER. THE ORIGINAL
00200 ; RELEASES OF THESE PROGRAMS ARE COPYRIGHTED BY OTHERS AND SOLD BY RADIO
00210 ; SHACK. THE USERS/BUYERS OF THIS UPDATED SYSTEM ARE ASSUMED TO HAVE PURCHASED
00220 ; THESE ORIGINAL PROGRAMS FROM RADIO SHACK, THEREBY OBTAINING THE ORIGINAL
00230 ; DOCUMENTATION AND IMPLICITLY PAYING
00240 ; THE ROYALTIES DUE. APPARAT'S REFUSAL TO SUPPLY COPIES OR EXTRACTS OF
00250 ; THE ORIGINAL DOCUMENTATION IS ITS SOLE EFFORT TO ASSURE ROYALTIES ARE
00260 ; BEING PAID.
00270 ;
00280 ; APPARAT, INC.
00290 ; 1. PROVIDES THIS SYSTEM SOLELY IN AN 'AS IS' CONDITION.
00300 ; 2. DOES NOT GUARANTEE THAT, AND PROBABLY DOES NOT KNOW WHETHER, ANY
00310 ; PARTICULAR PROGRAM/APPLICATION WILL FUNCTION PROPERLY OR
00320 ; SATISFACTORILY WITH THIS SYSTEM.
00330 ; 3. REQUIRES THAT THE USER THOROUGHLY TEST THIS SYSTEM BEFORE
00340 ; USING IT WITH VALUED DATA OR APPLICATIONS.
00350 ; 4. IS NOT RESPONSIBLE FOR DAMAGES CAUSED BY USE OF THIS SYSTEM.
00360 ; 5. DOES NOT PROVIDE MAINTENANCE, UPGRADE OR CONSULTATION SERVICES
00370 ; AS PART OF THIS OFFERING.
00380 ; 6. HOPES THE SYSTEM INFORMATION PROVIDED HEREIN IS ACCURATE BUT
00390 ; CANNOT GUARANTEE IT.
00400 ;
00410 ; *****
00420 ;
00430 ;
00440 ;
00450 ; EACH CHANGE IS HEADED BY A LINE OF ASTERISKS, CONTAINING THE CHANGE ID.
00460 ;
00470 ;
00480 ;
00490 ; THE HANDLING OF THE DIRECTORIES IS ESSENTIALLY THE SAME AS IN DOS 2.1.
00500 ;
00510 ; THIS SYSTEM CONSISTS OF THE FOLLOWING MODULES, WHICH IF NOT NOTED AS
00520 ; NEW, ARE SIMPLY THE ORIGINAL DOS 2.1 MODULES WITH CORRECTIONS AND ENHANCEMENTS:
00530 ; 0. DIR/SYS 2 GRANULES. DISKETTE DIRECTORY.
00540 ; 1. BOOT/SYS 1 GRANULE. BOOTSTRAPS THE SYSTEM.
00550 ; 2. SYS0/SYS 1 GRANULE. RESIDENT MODULE. HANDLES DISK I/O,
00560 ; CLOCK INTERRUPTS, LOAD OF SYSTEM OVERLAY
00570 ; MODULES, ETC.
00580 ; 3. SYS1/SYS 1 GRANULE. INTERROGATES DOS COMMANDS.
00590 ; 4. SYS2/SYS 1 GRANULE. OPEN EXECUTOR AND OTHER FUNCTIONS.
00600 ; 5. SYS3/SYS 1 GRANULE. HANDLES CLOSE AND KILL. CONTAINS COPY'S
00610 ; FORMATTING CODE.
00620 ; 6. SYS4/SYS 1 GRANULE. DISPLAYS SYSTEM ERROR MSGS.
00630 ; 7. SYS5/SYS 1 GRANULE. 'DEBUG' AND CMD"D" EXECUTOR.
00640 ; 8. SYS6/SYS 3 GRANULES. DOS COMMAND EXECUTOR, EXCEPT COPY AND DEBUG.
00650 ; 9. FORMAT/CMD 3 GRANULES. 'FORMAT' EXECUTOR.
00660 ; 10. COPY/CMD 1 GRANULE. NEW MODULE. EXECUTES DOS COPY COMMAND WITH
00670 ; SINGLE-DRIVE AND 'BACKUP' CAPABILITIES.
00680 ; 11. BASIC/CMD 4 GRANULES. EXECUTES BASIC'S LEVEL III FUNCTIONS.
00690 ; 12. SYS13/SYS 1 GRANULE. NEW MODULE. DISPLAYS BASIC'S ERROR

```

00700 #          MESSAGES. MUST BE IN SYSTEM WHENEVER BASIC
00710 #          IS ACTIVE.
00720 #          13. SYS12/SYS  1 GRANULE. NEW MODULE. EXECUTES BASIC DIRECT COMMAND
00730 #          'REF'. NEED NOT BE IN SYSTEM IF REF WILL
00740 #          NEVER BE EXECUTED.
00750 #          14. SYS11/SYS  1 GRANULE. NEW MODULE. EXECUTES BASIC DIRECT COMMAND
00760 #          'RENUM'. NEED NOT BE IN SYSTEM IF RENUM WILL
00770 #          NEVER BE EXECUTED.
00780 #          15. DIRCHECK/CMD 3 GRANULES. NEW MODULE. MAKES CHECKS AND LIST/
00790 #          PRINTS DIRECTORY CONTENTS.
00800 #          16. EDTASH/CMD  5 GRANULES. EDITOR-ASSEMBLER FOR 2-80 CODE. SOURCE
00810 #          AND OBJECT CODE FROM/TO DISK OR TAPE.
00820 #          17. DISKDUMP/BAS 2 GRANULES. DUMPS DISK FILES TO DISPLAY OR LINE PRINTER.
00830 #          18. BASIC1/CMD  PROVIDES A LEVEL-1-IN-LEVEL-II CAPABILITY.
00840 #          19. LVDSKSL/CMD  ALLOWS STORAGE AND RETRIEVAL OF LEVEL I PROGRAMS
00850 #          BUT NOT DATA TO/FROM DISK.
00860 #          20. SUPERZAP/BAS BASIC PROGRAM THAT ALLOWS INSPECTION AND MODIFICATION
00870 #          OF EITHER DISK OR MAIN MEMORY. DISK OPS ARE NOT
00880 #          FILE ORIENTED.
00890 #          21. SUPERZAP/COM BASIC PROGRAM SERVING AS SUPERZAP DOCUMENTATION.
00900 #
00910 #***** REDUCED-SIZE OPERATING SYSTEMS CAN BE CREATED BY 'COPY'ING FULL NEWDOS
00920 #DISKETTE, THEN 'KILL'ING UNWANTED FILES (SEE FOLLOWING).
00930 #
00940 #A MINIMUM SYSTEM TO HANDLE OPEN'S AND CLOSE'S WILL CONSIST OF 10 GRANULES
00950 #(ROOT, DIR, SYS0-SYS4)
00960 #IF DEBUG TO BE USED, ADD SYS5.
00970 #IF DOS COMMANDS (DIR, CLOCK, APPEND, DATE, ETC) TO BE EXECUTED, ADD SYS6
00980 #(SYS5 NOT REQUIRED).
00990 #IF DOS COMMAND COPY TO BE EXECUTED, ADD COPY/CMD. (SYS5, SYS6 NOT REQUIRED)
01000 #IF FORMAT TO BE EXECUTED, ADD FORMAT/CMD. (SYS5, SYS6, COPY NOT REQUIRED)
01010 #IF BASIC TO BE ENTERED, ADD BASIC/CMD AND SYS13. ONCE BASIC IS LOADED,
01020 #IT DOES NOT HAVE TO REMAIN PART OF THE SYSTEM DISKETTE CONTENTS (BUT
01030 #SYS13 DOES). SYS5 NOT NEEDED IF CMD"D" NOT USED. SYS6, COPY, FORMAT NOT
01040 #NEEDED UNLESS ASSOCIATED COMMANDS ARE EXECUTED VIA CMD"XX" FUNCTION.
01050 #IF BASIC DIRECT COMMAND REF TO BE EXECUTED, ADD SYS12.
01060 #IF BASIC DIRECT COMMAND RENUM TO BE EXECUTED, ADD SYS11.
01070 #
01080 #BASIC, COPY, FORMAT DO NOT HAVE TO RESIDE ON THE SYSTEM DISKETTE ON
01090 #DRIVE 0; BY APPENDING THE DRIVE # (1E, COPY:2) TO THE COMMAND NAME
01100 #THE MODULE CAN BE TAKEN FROM THE DISKETTE ON THE SPECIFIED DRIVE.
01110 #
01120 #
01130 #
01140 #***** THE SYSTEM CHANGES HEREIN MAKE THIS SYSTEM INCOMPATIBLE WITH THE
01150 #ORIGINAL DOS 2.1 SYSTEM. NO MODULE OF THIS SYSTEM (SYS0-SYS6, SYS11-SYS13,
01160 #COPY/CMD, BASIC/CMD OR FORMAT/CMD)
01170 #MAY BE USED INTERCHANGEABLY WITH MODULES FROM THE ORIGINAL DOS 2.1 OR
01180 #ANY OTHER SYSTEM, OR VICE-VERSA. CHANGING SYSTEMS MUST BE ACCOMPLISHED VIA
01190 #'RESET' OR 'POWER-ON' TO ASSURE THE PROPER RESIDENT MODULE (SYS0) LOAD.
01200 #WHEN COPYING FILES INTO OR OUT OF THIS SYSTEM, THE SOURCE AND/OR THE DESTINATION
01210 #FILE MAY RESIDE ON AN ALIEN SYSTEM DISKETTE PROVIDED ONE OF THE FOLLOWING EXISTS:
01220 #      1. THE ALIEN SYSTEM DISKETTE IS NEVER MOUNTED ON DRIVE ZERO.
01230 #      2. THE COPY COMMAND HAS THE SOURCE FILESPEC PRECEDED BY A $.
01240 #THE ORIGINAL DOS 2.1 'BACKUP' MODULE WILL WORK WITH THIS SYSTEM. HOWEVER,
01250 #AN OPTION OF THIS SYSTEM'S COPY ACCOMPLISHES ESSENTIALLY THE SAME THING.
01260 #*****
01270 #
01280 #USE OF 'COPY' IS DISCUSSED IN CHANGE # 040. ITS FORMAT #5 REPLACES 'BACKUP'.
01290 #USE OF 'REF' IS DISCUSSED IN CHANGE # 045.

```

01300 #USE OF 'RENUM' IS DISCUSSED IN CHANGE # 046.
01310 #USE OF 'FORMAT' IS DISCUSSED IN CHANGE # 037.
01320 #USE OF 'JKL' OPTION IS DISCUSSED IN CHANGE # 017.
01330 #KEYBOARD DEBOUNCE IS DISCUSSED IN CHANGE # 016.
01340 #NEW BASIC INVOCATION IS DISCUSSED IN CHANGE # 039.
01350 #BASIC'S CMD"XX" FUNCTION IS DISCUSSED IN CHANGE # 035.
01360 #BASIC SCROLLING COMMANDS DISCUSSED IN CHANGE # 031.
01370 #TRUNCATED BASIC LIST, EDIT AND DELETE COMMANDS ARE DISCUSSED IN CHANGES
01380 # # 047 AND 031.
01390 #BASIC OPEN"E" IS DISCUSSED IN CHANGE # 019.
01400 #DISK DUMP/BAS' IMPROVEMENTS DISCUSSED IN CHANGE # 048.
01410 #EDTASM/CMD' IMPROVEMENT DISCUSSED IN CHANGE # 049.
01420 #LEVEL I IN LEVEL II OPERATION. CHANGES # 050 AND 051.
01430 #
01440 #*****
01450 #
01460 #THE FOLLOWING SHOULD BE NOTED:
01470 # 1. THIS SYSTEM DOES NOT ADVANCE DATE AT 2400.
01480 # 2. ALL PASSWORD PROTECTION IS DISABLED. SEE CHANGE #009 FOR
01490 # PASSWORD ENABLE/DISABLE INSTRUCTIONS.
01500 # 3. DEVICE COMMAND IS DISABLED.
01510 # 4. ALL BASIC WRITES TO DISK, VIA PRINT AND PUT COMMANDS ARE VALIDITY
01520 # READ. USER HAS NO CONTROL OF THIS.
01530 # 5. ALL BASIC PROGRAMS 'SAVED' TO DISK ARE VALIDITY READ.
01540 # 6. ALL FILES DUPLICATED BY 'COPY' ARE VALIDITY READ.
01550 # 7. 'APPEND' NOW WORKS.
01560 # 8. FILE EOF'S ARE HANDLED CORRECTLY.
01570 # 9. BASIC'S 'LOC' COMMAND NOW WORKS PROPERLY.
01580 # 10. DOS CMD 'VERIFY' NOW WORKS PROPERLY.
01590 # 11. ALL SYSTEM DIRECTORY WRITES ARE NOW VALIDITY READ.
01600 # 12. BASIC 'LOAD' AND 'SAVE' RUN 0-30% FASTER, DEPENDING UPON
01610 # PROGRAM SIZE.
01620 # 13. DUE TO CORRECTIONS IN SYS3, A NUMBER OF DIRECTORY CLOBBERING
01630 # SITUATIONS HAVE BEEN ELIMINATED.
01640 # 14. DISK SPACE ALLOCATION IS 1 GRANULE INCREMENTS, INSTEAD OF 2.
01650 # THIS INCREASES
01660 # WRITE OVERHEAD BUT ALLOWS LAST GRANULE ON DISKETTE TO BE USED.
01670 # 15. BASIC'S OPEN"E" FUNCTION ALLOWS A BASIC PROGRAM TO ADD TO AN
01680 # EXISTING SEQUENTIAL FILE.
01690 # 16. THE DISK BACKUP FUNCTION OF SUPERZAP ALLOWS INFINITE RETRY
01700 # TO RECOVER INFORMATION FROM A BAD SECTOR WHEN 'BACKUP' OR
01710 # COPY (FULL DISKETTE OPTION) FAIL. IF THAT STILL FAILS, THEN
01720 # BYPASS THE BAD SECTOR(S) AND USE SUPERZAP'S 'SCOPY' OPTION
01730 # TO RECOVER AS MUCH AS POSSIBLE OF THE BAD SECTOR, MOVING IT
01740 # TO A DIFFERENT SECTOR WHERE IT POSSIBLY CAN BE CORRECTED.
01750 # 17. THE 'JKL' OPTION ALLOWS THE FULL DISPLAY CONTENTS TO BE SENT
01760 # TO THE LINE PRINTER, WITHOUT DISTURBING OR ADDING TO THE
01770 # DISPLAY CONTENTS.
01780 # 18. LEVEL1-IN-LEVEL2 OPERATION IS AVAILABLE THROUGH 'BASIC1' AND
01790 # 'LV1DISKSL' DOS COMMANDS. SEE CHANGES 050 AND 051.
01800 # 19. UNDER OLD DOS, IF NORMAL USER DISK I/O REFERENCES A DRIVE
01810 # WITHOUT A DISKETTE OR WITH DISKETTE IMPROPERLY MOUNTED, THE
01820 # SYSTEM MIGHT HANG. THIS SYSTEM WILL PROBABLY CATCH THE
01830 # CONDITION, THOUGH IT MIGHT TAKE 30 SECONDS.
01840 # 20. THE OCCURRENCE OF 'LOST DATA' DISK ERRORS IS GREATLY REDUCED.
01850 # 21. ON EVERY SECTOR READ/WRITE, THE CLOCK LOSES APPROXIMATELY
01860 # 120MS. IF THIS IS IMPORTANT, SEE CHANGE #007.
01870 #
01880 #*****
01890 #

01900 ;
01910 ;
01920 ;
01930 ;
01940 ;
01950 ;
01960 ;
01970 ;
01980 ;
01990 ;
02000 ;
02010 ;
02020 ;
02030 ;
02040 ;
02050 ;
02060 ;
02070 ;
02080 ;
02090 ;
02100 ;
02110 ;
02120 ;
02130 ;
02140 ;
02150 ;
02160 ;
02170 ;
02180 ;
02190 ;
02200 ;
02210 ;
02220 ;
02230 ;
02240 ;
02250 ;
02260 ;
02270 ;
02280 ;
02290 ;
02300 ;
02310 ;
02320 ;
02330 ;
02340 ;
02350 ;
02360 ;
02370 ;
02380 ;
02390 ;
02400 ;
02410 ;
02420 ;
02430 ;
02440 ;
02450 ;
02460 ;
02470 ;
02480 ;
02490 ;

DOS NOMENCLATURE AND STRUCTURE:

1. SECTOR UNIT OF 256 BYTES STORED ON DISK.
2. GRANULE UNIT OF 5 SECTORS. THE BASIC UNIT OF DISK STORAGE ALLOCATION. DIRECTORIES KEEP TRACK OF FREE/ASSIGNED DISK SPACE IN TERMS OF GRANULES.
3. TRACK UNIT OF 10 SECTORS (2 GRANULES); EQUALS THE # OF DATA BYTES THE DISK READ/WRITE HEAD PASSES OVER ON EACH DISKETTE REVOLUTION.
4. BOOT/SYS A FILE PLACED ON EVERY DISKETTE BY 'FORMAT'. THIS IS THE BOOTSTRAP ROUTINE INVOKED WHEN RESET OR POWER-ON OCCURS. IT AUTOMATICALLY LOOKS IN THE DIRECTORY FOR SYS0/SYS'S ENTRY. IF PRESENT AND ACTIVE, SYS0/SYS IS LOADED INTO MEMORY AS THE DOS RESIDENT MODULE. OTHERWISE THE SYSTEM HANGS.
5. DIR/SYS A FILE PLACED ON EVERY DISKETTE BY FORMAT; IT IS REQUIRED FOR PROPER DOS OPERATION. CURRENTLY IT CONSISTS OF 10 SECTORS USED AS FOLLOWS:
 1. SECTOR 0 (GAT SECTOR) CONTAINS:
 1. DISKETTE NAME. (OFFSET = D0-D7 HEX)
 2. LATEST CREATION, BACKUP OR COPY DATE. (OFFSET = D8-DF HEX)
 3. DISKETTE PASSWORD ENCODE. (OFFSET = CE-CF HEX)
 4. DOS 'AUTO' INFORMATION. (OFFSET E0-FF HEX)
 5. TRACK LOCKOUT INFO. OFFSET 60 HEX AND UP FOR AS MANY BYTES AS THE DISKETTE HAS TRACKS. FC HEX VALUE MEANS TRACK NOT LOCKED OUT; FF HEX VALUE MEANS TRACK IS LOCKED OUT.
 6. GRANULE FREE/ASSIGNED STATUS. OFFSET 00 HEX AND UP FOR AS MANY BYTES AS THE DISKETTE HAS TRACKS. EACH BYTE CONTAINS INFORMATION FOR THE TWO GRANULES ON THAT TRACK.
1ST 6 BITS SHOULD ALWAYS = 1. 7TH BIT IS FOR THE 2ND GRANULE OF THE TRACK (SECTORS 5-9). THE 8TH BIT IS FOR THE 1ST GRANULE OF THE TRACK (SECTORS 0-4). FOR THESE TWO BITS, THE CORRESPONDING GRANULE IS FREE IF THE BIT = 0; IS ALLOCATED TO A FILE IF = 1.
 2. SECTOR 1 (HIT SECTOR) CONTAINS A 1 BYTE HASH CODE FOR EACH ACTIVE DIRECTORY ENTRY (4TH BIT, 1ST BYTE = 1) IN SECTORS 2-9, THUS ALLOWING RAPID ACCESS TO FILES. THE HASH CODE IS BASED ON THE CONTENTS OF FPDE, BYTES 6 - 16.
THE RELATIVE POSITION IN THE HIT SECTOR OF A DIRECTORY ENTRY'S HASH CODE IS EQUAL TO THAT ENTRY'S DEC. THE OTHER 192 BYTES OF THE SECTOR ARE NOT CURRENTLY USED AND SHOULD ALWAYS = 0.
 3. SECTORS 2-9 EACH CONTAIN SPACE FOR EIGHT 32 BYTE DIRECTORY ENTRIES (FPDE OR FXDE). 1ST TWO ENTRIES OF EACH SECTOR ARE RESERVED FOR THE SYSTEM.
NORMALLY THE DIRECTORY IS LOCATED ON DISKETTE'S RELATIVE TRACK 11 HEX.
6. FPDE FILE PRIMARY DIRECTORY ENTRY. EACH FILE, WHEN CREATED, IS ASSIGNED A DIRECTORY ENTRY SOMEWHERE IN DIRECTORY SECTORS 2-9. THIS ENTRY CONTAINS:
 1. 1ST BYTE, 1ST BIT = 0, INDICATING FPDE, VICE FXDE.
 2. 1ST BYTE, 2ND BIT = 1 IF A SYSTEM FILE.
 3. 1ST BYTE, 4TH BIT = 1 IF THIS ENTRY IS ASSIGNED TO A

	FREE	
	0-4	5-9
C	X	X
D	U	X
E	X	U
F	U	U

02500 ; FILE AND HAS A NON-ZERO HASH CODE IN THE HIT SECTOR.
02510 ; THE OFFSET IN THE HIT SECTOR OF THIS ENTRY'S HASH
02520 ; CODE = THIS ENTRY'S DEC.
02530 ; IF THIS BIT = 0, THE DIRECTORY ENTRY IS AVAILABLE
02540 ; FOR REASSIGNMENT; I.E., IF FIRST HEX DIGIT ON
02550 ; SUPERZAP'S DUMP LINE IS 0.
02560 ; 4. 1ST BYTE, 5TH BIT = 1 IF A FILE HAS THE INVISIBLE ATTRIBUTE.
02570 ; 5. 1ST BYTE, 6-8TH BITS = PROTECTION LEVEL:
02580 ; 7 = NO ACCESS.
02590 ; 6 = EXECUTION ACCESS ONLY.
02600 ; 5 = READ, EXECUTE ACCESSES
02610 ; 4 = WRITE, READ, EXECUTE
02620 ; 3 = (UNUSED)
02630 ; 2 = RENAME, WRITE, READ, EXECUTE
02640 ; 1 = KILL, RENAME, WRITE, READ, EXECUTE
02650 ; 0 = NO RESTRICTIONS
02660 ; 6. 2ND BYTE = 0, USE UNKNOWN.
02670 ; 7. 3RD BYTE = 0, USE UNKNOWN.
02680 ; 8. 4TH BYTE = 0-255 END-OF-FILE RELATIVE POSITION IN THE EOF
02690 ; SECTOR.
02700 ; 9. 5TH BYTE = 0-255 LOGICAL RECORD LENGTH. 0 = 256.
02710 ; WHEN FILE IS CREATED VIA A 4420H VECTOR CALL, THE
02720 ; VALUE FROM REGISTER B IS STORED HERE. WHEN AN EXISTING
02730 ; FILE IS OPENED, EVEN AS A NEW OUTPUT FILE, THIS VALUE
02740 ; IS NOT UPDATED. THIS VALUE IS NEVER USED. THE
02750 ; VALUE STORED IN FCB49 AT OPEN TIME IS THAT FROM
02760 ; REGISTER B, NOT FROM THE FPDE.
02770 ; 10. 6TH-13TH BYTES = FILE NAME, PADDED ON RIGHT WITH
02780 ; BLANKS IF NECESSARY.
02790 ; 11. 14TH-16TH BYTES = FILE NAME MODIFIER, PADDED ON RIGHT
02800 ; WITH BLANKS AS NECESSARY.
02810 ; 12. 17TH-18TH BYTES = ENCODE OF UPDATE PASSWORD.
02820 ; 13. 19TH-20TH BYTES = ENCODE OF ACCESS PASSWORD.
02830 ; 14. 21ST-22ND BYTES = RELATIVE SECTOR # WITHIN THE FILE
02840 ; OF EOF. HOWEVER, IF THE 4TH BYTE IS NON-ZERO,
02850 ; THIS VALUE IS RELATIVE EOF SECTOR + 1. THIS SYSTEM
02860 ; MAINTAINS THE EOF WITHIN THE FPDE COMPATIBLE WITH
02870 ; THAT OF TRSDOS 2.1 SO THAT A FILE MAY BE USED
02880 ; INTERCHANGEABLY BETWEEN THE TWO SYSTEMS.
02890 ; WHEN A FILE IS OPEN, THE EOF VALUE IN
02900 ; THE FCB IS MAINTAINED DIFFERENTLY IN THE TWO SYSTEMS.
02910 ; 15. 23RD-30TH BYTES = FOUR 2 BYTE PAIRS (EXTENT ELEMENTS),
02920 ; EACH SPECIFYING A CONTIGUOUS AREA OF THE DISKETTE
02930 ; ASSIGNED TO THIS FILE. THE FORMAT OF AN EXTENT
02940 ; ELEMENT IS:
02950 ; 1. 1ST BYTE:
02960 ; 1. 255 = END OF EXTENTS.
02970 ; 2. 254 = 2ND BYTE CONTAINS THE DEC
02980 ; FOR THE NEXT FXDE ASSIGNED
02990 ; TO THIS FILE TO CONTAIN ADDITIONAL
03000 ; EXTENT INFORMATION.
03010 ; 3. 0 - 34 = RELATIVE TRACK ON DISKETTE FOR
03020 ; START OF AREA. IF YOUR DISKETTES
03030 ; HAVE MORE THAN 35 TRACKS, THIS VALUE
03040 ; CAN COVER 0 - 253 RANGE.
03050 ; 2. 2ND BYTE (WHEN 1ST BYTE < 254)
03060 ; 1. LEFT 3 BITS = # GRANULES (0-7) OFFSET
03070 ; FROM START OF TRACK TO START OF AREA.
03080 ; 2. RIGHT 5 BITS = #-1 OF CONTIGUOUS
03090 ; GRANULES ASSIGNED TO THIS AREA.

03100 ; 16. 31-32ND BYTES CONTAIN AN EXTENT ELEMENT WHOSE 1ST BYTE
03110 ; IS EITHER 255 OR 254.

03120 ; 7. FXDE FILE EXTENDED DIRECTORY ENTRY. WHEN A FILE HAS
03130 ; MORE THAN 4 SPACE AREAS ASSIGNED, THE ADDITIONAL EXTENT
03140 ; ELEMENTS ARE CONTAINED IN FXDE'S ASSIGNED TO THE FILE.
03150 ; THE FORMAT OF A FXDE IS:
03160 ; 1. 1ST BYTE, 1ST BIT = 1 TO INDICATE FXDE, VICE FXPE.
03170 ; 2. 1ST BYTE, 4TH BIT. SEE FPDE.
03180 ; 3. 2ND BYTE = DEC FOR PREVIOUS FXDE OR FPDE
03190 ; OF THIS FILE. THIS IS A BACKWARD CHAIN.
03200 ; THE PREVIOUS ENTRY'S 31ST BYTE WILL BE 254, AND
03210 ; THE 32ND BYTE WILL CONTAIN THE DEC OF THIS FXDE.
03220 ; 4. BYTES 3-22 ARE UNUSED AND SHOULD = 0.
03230 ; 5. BYTES 23-32 ARE AS DEFINED FOR FPDE.

03240 ; 8. EXTENT ELEMENT (SEE FPDE, BYTES 23-30).

03250 ; 9. DEC DIRECTORY ENTRY CODE. A ONE BYTE CODE USED TO
03260 ; SPECIFY ONE OF THE 64 DIRECTORY ENTRIES IN DIRECTORY
03270 ; SECTORS 2 - 9. THE FORMAT IS:
03280 ; RRR00SSS WHERE
03290 ; SSS+2 = RELATIVE SECTOR IN DIRECTORY
03300 ; RRR = RELATIVE 32 BYTE DIRECTORY ENTRY IN THAT SECTOR.

03310 ; 10. GAT GRANULE ALLOCATION TABLE. REFERS TO DIRECTORY RELATIVE
03320 ; SECTOR 0.

03330 ; 11. HIT HASH INDEX TABLE. REFERS TO DIRECTORY RELATIVE SECTOR 1.

03340 ; 12. VICE MEANS 'INSTEAD OF' OR 'IN PLACE OF'.

03350 ; 13. FCB FILE CONTROL BLOCK. ALSO KNOWN AS AN IOB
03360 ; (INPUT/OUTPUT BLOCK). AT OPEN TIME (CALL TO DOS 4420H
03370 ; OR 4424H), THE CALLER PROVIDES IN REGISTER DE THE ADDRESS OF
03380 ; A 32 BYTE MAIN MEMORY AREA FOR USE BY THE SYSTEM WHILE THE
03390 ; FILE IS OPEN. THE USER MUST HAVE PLACED THE FILE-SPEC INTO
03400 ; THE FCB'S 1ST BYTES, AND CLOSE WILL ATTEMPT TO PUT IT
03410 ; BACK THERE WHEN DONE. WHILE THE FILE IS OPEN, THE FORMAT FOR
03420 ; THE 32 BYTE FCB IS:
03430 ; 1. 1ST BYTE, 1ST BIT = 1, FILE IS OPEN; = 0, FILE IS CLOSED.
03440 ; 2. 2ND BYTE, 1ST BIT = 1, EITHER SINGLE BYTE OPERATIONS OR
03450 ; LOGICAL RECORD OPERATIONS (REC LENGTH IN 10TH BYTE)
03460 ; BEING DONE ON THIS FILE. 'NEXT' VALUE IS MAINTAINED
03470 ; AT THE NEXT BYTE TO BE READ/Written.
03480 ; = 0, SECTOR OPERATIONS; CALLS TO SYSTEM ARE BY
03490 ; FULL 256 BYTE SECTOR. 'NEXT' VALUE IS MAINTAINED
03500 ; AT START OF NEXT SECTOR TO BE READ/Written.
03510 ; THIS BIT IS SET = 1 AT OPEN TIME IF REGISTER B
03520 ; NOT = 0. IT IS ALSO SET = 1 WHENEVER ONE OF THE
03530 ; ROM (LEVEL II BASIC) SINGLE BYTE READ/WRITE ROUTINES
03540 ; ARE CALLED WITH REGISTER DE CONTAINING THE FCB ADDR.
03550 ; 3. 2ND BYTE, 2ND BIT
03560 ; = 0, EOF IS TO BE SET = TO 'NEXT' ON EVERY WRITE OPERATION.
03570 ; = 1, EOF IS TO BE SET = TO 'NEXT' ONLY FOR THOSE
03580 ; WRITE OPERATIONS RESULTING IN 'NEXT'
03590 ; EXCEEDING THE OLD EOF.

03600 ; 4. 2ND BYTE, 3RD BIT
03610 ; = 0, BUFFER CONTAINS THE CURRENT SECTOR.
03620 ; = 1, BUFFER DOES NOT CONTAIN THE CURRENT SECTOR.

03630 ; 5. 2ND BYTE, 4TH BIT
03640 ; = 0, BUFFER DOES NOT CONTAIN UPDATED DATA NOT
03650 ; YET SENT TO THE FILE.
03660 ; = 1, BUFFER DOES CONTAIN UPDATED DATA NOT YET
03670 ; SEND TO THE FILE.
03680 ; AT CLOSE TIME, IF THIS BIT AND 2ND BYTE, 1ST BIT
03690 ; BOTH = 1, THE RESIDUAL SECTOR IN THE BUFFER IS

03700 ;
03710 ;
03720 ;
03730 ;
03740 ;
03750 ;
03760 ;
03770 ;
03780 ;
03790 ;
03800 ;
03810 ;
03820 ;
03830 ;
03840 ;
03850 ;
03860 ;
03870 ;
03880 ;
03890 ;
03900 ;
03910 ;
03920 ;
03930 ;
03940 ;
03950 ;
03960 ;
03970 ;
03980 ;
03990 ;
04000 ;
04010 ;
04020 ;
04030 ;
04040 ;
04050 ;
04060 ;
04070 ;
04080 ;
04090 ;
04100 ;
04110 ;
04120 ;
04130 ;
04140 ;
04150 ;
04160 ;
04170 ;
04180 ;
04190 ;
04200 ;
04210 ;
04220 ;
04230 ;
04240 ;
04250 ;
04260 ;
04270 ;
04280 ;
04290 ;

AUTOMATICALLY WRITTEN TO DISK.

6. 2ND BYTE, 6-8TH BITS = PROTECTION CODE (SEE FPDE, 1ST BYTE 6-8TH BITS) IF PASSWORD CHECKING ENABLED AND ACCESS, VICE UPDATE, PASSWORD SPECIFIED IN THE FILESPEC AT OPEN TIME.

7. 3RD BYTE, SET = 0 AT OPEN TIME. DON'T KNOW WHAT IT'S USED FOR.

8. 4-5TH BYTES CONTAIN THE BUFFER ADDRESS SUPPLIED IN REG HL BY CALLER AT OPEN TIME.

9. 6TH BYTE LOW ORDER BYTE OF 3 BYTE 'NEXT' VALUE. THIS IS THE RELATIVE POSITION WITHIN SECTOR VALUE. SEE DISCUSSION FOR 11-12TH BYTES BELOW.

10. 7TH BYTE RELATIVE DRIVE # OF DRIVE CONTAINING DISKETTE CONTAINING THE FILE AT OPEN TIME.

11. 8TH BYTE DEC OF FILE'S FPDE.

12. 9TH BYTE LOW ORDER BYTE OF EOF. SEE DISCUSSION OF FCB, 13-14TH BYTES.

13. 10TH BYTE 0-256 LOGICAL RECORD LENGTH (LRL) FOR LOGICAL RECORDS OF THIS FILE, 0 = 256. THIS VALUE IS SUPPLIED IN REGISTER B BY THE CALLER AT OPEN TIME. IF NOT = 0 AT OPEN TIME, FCB 1ST BYTE, 1ST BIT IS SET = 1, AND DOS SECTOR READ/WRITE CALLS MUST CONTAIN IN REGISTER HL THE ADDRESS OF THE LOGICAL RECORD TO BE MOVED TO THE FCB'S BUFFER (WRITE) OR FILLED FROM THE FCB'S BUFFER (READ).

14. 11-12TH BYTES 12TH, 11TH AND 5TH BYTES FORM A 3 BYTE RELATIVE OFFSET IN FILE TO THE NEXT BYTE TO BE PROCESSED, EITHER INPUT OR OUTPUT. FOR BYTES OPERATIONS (2ND BYTE, 1ST BIT = 1) THIS SYSTEM MAINTAINS THIS 3 BYTE VALUE EXACTLY WHEREAS THE ORIGINAL DOS SYSTEM MAINTAINED THIS VALUE HIGH BY 256, CAUSING SOME PROBLEMS IN 'COPY'.
FOR FULL SECTOR OPERATIONS, BOTH THE ORIGINAL DOS AND THIS SYSTEM ADVANCE 'NEXT' BY 256 ON EACH SUCCESSFUL READ OR WRITE TO DISK, EXCEPTING THAT IN THIS SYSTEM, 'NEXT' IS NOT ADVANCED ON FULL SECTOR WRITE IF THE FCB'S 6TH BYTE NOT = 0.

15. 13-14TH BYTES. THE 14TH, 13TH AND 8TH BYTES FORM 3 BYTE RELATIVE BYTE OFFSET IN THE FILE OF END-OF-FILE (THE 1ST BYTE BEYOND THE FILE'S LAST DATA BYTE). THIS VALUE IS INITIALIZED FROM THE FPDE AT OPEN TIME, AND IS UPDATED AT SECTOR, LOGICAL RECORD OR BYTE WRITE TIME UNDER CONTROL OF FCB, 2ND BYTE, 2ND BIT. THIS SYSTEM MAINTAINS THE EOF IN THE FCB EXACTLY CURRENT WHILE THE ORIGINAL DOS SYSTEM MAINTAINED IT HIGH BY 256 WHENEVER FCB 6TH BYTE NOT = 0.
FOR COMPABILITY, THIS SYSTEM, AT CLOSE TIME WHEN THE EOF VALUE IN THE FPDE IS UPDATED, MAKES THE EOF IN THE FPDE EXACTLY AS FOR THE ORIGINAL DOS SYSTEM.

16. 15-16TH BYTES. IDENTICAL TO 23-24TH BYTES OF FPDE.

17. 17-32TH BYTES. FOUR 4 BYTE ELEMENTS OF THE FORM:

1. 1-2ND BYTES. RELATIVE GRANULE OFFSET WITHIN THE FILE TO THE START OF THE DISK AREA WHOSE EXTENT ELEMENT FOLLOWS.

2. 3-4TH BYTES. THE EXTENT ELEMENT OF ONE OF THE AREAS ASSIGNED TO THE FILE (EXCEPT THE FIRST).
SEE FPDE, BYTES 23-30.

THESE FOUR ELEMENTS ALLOW THE SYSTEM TO REDUCE DIRECTORY ACCESS DURING RANDOM FILE OPERATIONS BY REMEMBERING THE MOST RECENT 4 AREAS (OTHER THAN THE


```
04300 ; FIRST) ACCESSED, UNDER THE ASSUMPTION THAT A SECTOR
04310 ; FROM ONE OF THOSE AREAS WILL BE THE NEXT ACCESSED OR
04320 ; WRITTEN.
04330 ; 14. EOF END-OF-FILE. THE RELATIVE BYTE ADDRESS+1 WITHIN
04340 ; THE FILE OF THE LAST BYTE OF THE FILE.
04350 ;
04360 ; END SYSMODS1/COM; APPEND SYSMODS2/COM NEXT!
04370 ; SYSMODS2/COM
04380 ;
04390 ;*** 001 ****
04400 ;
04410 ;DOS COMMAND 'DIRCHECK'
04420 ;
04430 ; THE DIRCHECK/CMD MODULE TESTS AND LISTS THE TARGET DISKETTE'S DIRECTORY.
04440 ;
04450 ; TO THE QUERY 'OUTPUT TO PRINTER', REPLY Y FOR YES OR N FOR NO (OUTPUT TO DISPLAY).
04460 ;TO THE QUERY 'WHICH DRIVE CONTAINS TARGET DISKETTE' REPLY A DECIMAL DIGIT 0 TO 4.
04470 ;
04480 ;COMPLAINTS ABOUT THE DIRECTORY ARE LISTED FIRST. IF A NUMBER IS GIVEN,
04490 ; IT IS IN HEXIDECIMAL FOR USE IN DIRECTORY REPAIR VIA 'SUPERZAP'.
04500 ;DO NOT TRY TO REPAIR A BAD DIRECTORY UNLESS YOU KNOW WHAT YOU ARE DOING!!!!!!
04510 ;THE NEXT BEST THING IS TO TRY TO EXTRACT VALUED FILES VIA COPY AND
04520 ;THEN RE-FORMAT THE DISKETTE HAVING THE BAD DIRECTORY.
04530 ;
04540 ;IF THE COMPLAINT IS ABOUT A DIRECTORY ENTRY FOR A FILE, EITHER THE PRIMARY
04550 ; OR AN EXTENDED ENTRY, THE HEXIDECIMAL CODE IS THE DEC
04560 ;FOR THE FILE'S FPDE.
04570 ;WHEN THE COMPLAINT DEALS WITH A FILE EXTENDED DIRECTORY ENTRY BUT DOES NOT
04580 ;SPECIFY THE FILE NAME/TYPE, THE HEXIDECIMAL CODE IS THE DEC
04590 ;FOR THE FXDE ITSELF.
04600 ;WHEN THE COMPLAINT DEALS WITH A HIT SECTOR BYTE, THE HEXIDECIMAL CODE IS THE
04610 ;RELATIVE ADDR OF THAT BYTE IN THE HIT SECTOR.
04620 ;WHEN THE COMPLAINT DEALS WITH A GAT SECTOR BYTE, THE HEXIDECIMAL CODE IS THE
04630 ;RELATIVE ADDR OF THAT BYTE IN THE GAT SECTOR.
04640 ;WHEN THE COMPLAINT DEALS WITH A GRANULE, THE HEXIDECIMAL VALUE IS THE RELATIVE
04650 ;GRANULE NUMBER WITHIN THE DISKETTE.
04660 ;WHEN THE COMPLAINT DEALS WITH A TRACK, THE HEXIDECIMAL VALUE IS THE RELATIVE
04670 ;TRACK NUMBER WITHIN THE DISKETTE.
04680 ;'DIRCHECK' ALSO DEMANDS THAT THE 1ST 2 BYTES OF THE DISKETTE'S 1ST SECTOR
04690 ;(BOOT/SYS'S 1ST SECTOR) CONTAIN 00 HEX AND FE HEX RESPECTIVELY. THE 3RD
04700 ;BYTE MUST CONTAIN THE RELATIVE # OF THE TRACK CONTAINING THE DIRECTORY.
04710 ;
04720 ;
04730 ; THE FILES ARE NEXT LISTED, WITH NUMERIC VALUES IN DECIMAL.
04740 ; 1. S SYSTEM FILE.
04750 ; 2. I FILE HAS INVISIBLE ATTRIBUTE.
04760 ; 3. P=NNN FILE HAS PROTECTION LEVEL NNN, AND BOTH UPDATE AND
04770 ; ACCESS PASSWORDS ARE NON-BLANK.
04780 ; 4. EOF = SSS/BBB SSS = THE RELATIVE SECTOR WITHIN THE FILE.
04790 ; BBB = THE RELATIVE BYTE WITHIN THE SECTOR.
04800 ; 5. NNN EXTS THE # OF EXTENT ELEMENTS WITH 1ST BYTE < 254.
04810 ; 6. NNN SECTORS THE NUMBER OF SECTORS ALLOCATED TO THIS FILE.
04820 ;
04830 ;TO THE QUERY 'START PROGRAM ANEW?', REPLY Y FOR YES OR N FOR NO (EXITS TO DOS).
04840 ;
04850 ; DURING DISPLAY/LISTING, PRESSING:
04860 ; 'BREAK' - DISPLAYING/LISTING WILL PAUSE AT END OF CURRENT LINE
04870 ; OR LINE GROUP.
04880 ; 'ENTER' - RESTARTS PAUSED DISPLAYING/LISTING.
04890 ; 'UP-ARROW' - TERMINATE DISPLAYING/LISTING.
```

```
04900 ;
04910 ;
04920 ;
04930 ;
04940 ;*** 002 ****
04950 ;
04960 ;CHANGE TO CAUSE DISK SEEK AND RESTORES TO CONSTANTLY RESELECT THE
04970 ;CURRENT DRIVE WHILE WAITING FOR DISK CONTROLLER TO CONCLUDE THE SEEK/RESTORE.
04980 ;IN THE ORIGINAL SYSTEM, THERE IS INSUFFICIENT DRIVE 'READY' TIME TO
04990 ;MOVE THE ARM FROM TRACK 00H TO TRACK 22H AND HAVE A FULL 200MS TO READ
05000 ;THE FARTHEST SECTOR ON THE TRACK. THIS IS DUE TO 'READY' BEING ON ONLY
05010 ;APPROX 2.4 SECS FROM LAST SELECT (WRITE TO 37E1H). THE TIME EXPIRES AFTER
05020 ;A 1 SECOND WAIT FOR DRIVE DEFAULT-UP-TO-SPEED AND THEN 34 40MS STEPS OF
05030 ;THE DISK ARM.
05040 ;DOS-BASIC 'GET' AND 'PUT' AVOIDED THIS PROBLEM BY DOING AN EXTRA SEEK
05050 ;VIA CALL AT 42ABH
05060 ;ANYWAY, THIS CHANGE KEEPS THE DRIVE READY FOR THE ENTIRE SEEK/RESTORE (EVEN
05070 ;IF THE ARM MUST STEP 70 TRACKS) PLUS APPROX 2.4 SECOND BEYOND THAT.
05080 ;SYS0 4869 AND OTHERS
05090 ;SYS3, FORMAT PART
05100 ;FORMAT 575C, 5764, 556D
05110 ;
05120 ;
05130 ;
05140 ;*** 003 ****
05150 ;
05160 ;ENABLES DEVICE-NOT-READY CONDITION TO BECOME A VALID ERROR IN SYS0'S READ/WRITE
05170 ;SECTOR ROUTINES, WHEREAS BEFORE IT WAS GOING UNNOTICED AND CAUSING CALLING
05180 ;ROUTINES TO ASSUME OPERATION COMPLETED WITHOUT ERROR.
05190 ; SYS0 46AF
05200 ;***** INCORPORATED IN CHANGE #007
05210 ;
05220 ;
05230 ;
05240 ;*** 004 ****
05250 ;
05260 ;DIRECT ADDRESSING, VICE INDIRECT, REQUIRED HERE.
05270 ;THIS ERROR CAUSES REAL TROUBLE FOR PROGRAMS THAT WANT TO USE 'BREAK' AS
05280 ;IT CAUSES THE SYSTEM TO THINK 'DEBUG' IS ACTIVATED WHEN IT HAS NOT BEEN.
05290 ;'EDTASH' MUST RUN WITH INTERRUPTS OFF BECAUSE OF THIS.
05300 ; SYS1, 4E28
05310 ;
05320 ;
05330 ;
05340 ;*** 005 ****
05350 ;
05360 ;CODE FOR SHIFTING DOWN OLDER USED EXTENT REFERENCES IN THE FCB IS JUST PLAIN
05370 ;WRONG.
05380 ;SYS0, 497A
05390 ;
05400 ;
05410 ;
05420 ;*** 006 ****
05430 ;
05440 ;NC STATE ALWAYS SET HERE. USE NZ INSTEAD, WHICH MEANS >= EOF.
05450 ;SYS0, 47F9
05460 ;***** INCORPORATED IN CHANGE #015
05470 ;
05480 ;
05490 ;
```

```

05500 *** 007 ****
05510 ;
05520 ; CHANGE TO:
05530 ;     1. FORCE VALIDITY READ AFTER EVERY SECTOR WRITE WHEN:
05540 ;         1. BIT 7, FCB 2ND BYTE = 1 (BYTE OPS)
05550 ;         2. BASIC IS WRITING TO DATA FILES
05560 ;         3. ALL SYSTEM WRITES NOT FOLLOWED BY A SUBSEQUENT READ.
05570 ;     2. DISABLE INTERRUPTS DURING SECTOR READ/WRITE OPERATIONS. THIS
05580 ;         CAUSES THE CLOCK TO LOSE APPROX 120MS EACH I/O.
05590 ;         IF A MORE ACCURATE CLOCK IS REQUIRED FOR USER OPERATIONS, THIS
05600 ;         CAN BE REDUCED TO ABOUT A 10MS CLOCK LOSS PER I/O WHILE HAVING
05610 ;         I/O'S TAKE 5-10% LONGER BY CHANGING SYS0, 4692.
05620 ;         SYS0, 4692 HEX IS LOCATED AS RELATIVE BYTE 4A HEX OF RELATIVE
05630 ;         SECTOR 4 OF FILE SYS0/SYS, AND IF FILE SYS0/SYS'S FPDE
05640 ;         23-24TH BYTES = 0022 HEX, THEN THE TARGET BYTE IS RELATIVE BYTE
05650 ;         4A HEX IN RELATIVE SECTOR 9 OF RELATIVE TRACK 0 ON THE DISKETTE.
05660 ;         THE BYTE PRECEDING THE TARGET BYTE = 37 HEX, AND THE BYTE
05670 ;         THE BYTE FOLLOWING = 36 HEX.
05680 ;         1. SET TARGET BYTE = 00 HEX IF A MORE ACCURATE CLOCK BUT
05690 ;             SLOWER I/O IS WANTED.
05700 ;         2. SET TARGET BYTE = F3 HEX IF CLOCK VALUE NOT IMPORTANT
05710 ;             AND THE FASTER I/O WANTED. THIS SYSTEM IS INITIALLY
05720 ;             SET TO THIS STATE.
05730 ;             AFTER CHANGING THIS BYTE, DO 'RESET'.
05740 ;     3. CATCH DEVICE-NOT-READY EVEN THOUGH CONTROLLER REMAINS BUSY.
05750 ;     4. DELAY RE-ENABLING INTERRUPTS UNTIL TERMINATING CONDITIONS HAVE
05760 ;         BEEN EXTRACTED FROM CONTROLLER. DON'T KNOW IF THIS WAS AN ERROR,
05770 ;         BUT IT SEEMS SENSIBLE.
05780 ; ***** HOPEFULLY, THIS WILL REDUCE DISK ERRORS.....
05790 ; SYS0, 4860, 4671, 468F, 45BF
05800 ; SYS2 514B
05810 ; SYS3 5014
05820 ; SYS4 5445
05830 ; BASIC 62BF
05840 ;
05850 ;
05860 ;
05870 *** 008 ****
05880 ;
05890 ; DISABLE ALL PASSWORD CHECKS IN 'OPEN'.
05900 ; PROTECTION CODE IS DISABLED. PROGRAMS OPERATE AS IF THEY HAD NO PASSWORDS
05910 ; AT ALL.
05920 ; THIS IS NOT AN ERROR CORRECTION BUT A CONVENIENCE.
05930 ; THIS PATCH NEEDED WHEN ACCESS/UPDATE NEEDED TO SYSTEM PROGRAMS BY METHOD OTHER
05940 ; THAN 'SUPERZAP'.
05950 ; ONLY ONE BYTE IS CHANGED, THAT LOCATED AT SYS2, 4EAC, WITH PASSWORD
05960 ; CHECKING ENABLED, THAT BYTE = 28 HEX. WITH PASSWORD CHECKING DISABLED
05970 ; THE BYTE = 18 HEX. THIS BYTE CAN BE ALTERED VIA 'SUPERZAP'. THE BYTE IS
05980 ; LOCATED IN FILE SYS2/SYS, RELATIVE SECTOR 1, RELATIVE BYTE 63 HEX. IF
05990 ; FILE SYS2/SYS HAS ITS FPDE, 23-24TH BYTES = 1020 HEX, THEN THE BYTE IS
06000 ; AT RELATIVE POSITION 63 HEX WITHIN RELATIVE SECTOR 6 WITHIN RELATIVE TRACK
06010 ; 10 HEX OF THE SUBJECT DISKETTE. AS A CHECK, THE PRECEDING BYTE =
06020 ; E1 HEX AND THE FOLLOWING BYTE = 21 HEX.
06030 ; SYS2, 4EAC
06040 ;
06050 ;
06060 ;
06070 *** 009 ****
06080 ;
06090 ; SYS4: B AND C REGS LOADED TOO LATE FOR DIRECTORY SECTOR READ AT 4EC4H.

```

06100 ; SYS4, 4E86
06110 ;
06120 ;
06130 ;
06140 ;*** 010 ****
06150 ;
06160 ; SYS0: PROTECTION CHECK DURING LOAD/EXEC FUNCTION IS THAT FOR 'READ', NOT 'EXEC'
06170 ; ALSO ENABLE CALL OF FILE-ACCESS-INHIBITED ERROR FOR LOAD/EXEC REGARDLESS
06180 ; OF 'DEBUG' STATUS.
06190 ; SYS0, 4C2E
06200 ;
06210 ;
06220 ;
06230 ;*** 011 ****
06240 ;
06250 ; HAVE SYS0 ALLOCATE IN UNITS OF 1 GRANULE, VICE 2, WHICH ALLOWS USE OF
06260 ; LAST GRANULE ON DISKETTE.
06270 ; THIS WILL INCREASE WRITE OVERHEADS.
06280 ; SYS0, 4A17
06290 ;
06300 ;
06310 ;
06320 ;*** 012 ****
06330 ;
06340 ; CHANGE TO ALLOW 'CLOSE' FUNCTION IN SYS3 TO DEALLOCATE THE UNUSED GRANULE WHEN
06350 ; THE LOGICAL END OF FILE IS THE FIRST BYTE OF THE GRANULE.
06360 ; THIS ASSUMES THAT THE EOF BYTE IS NOT A PART OF THE FILE.
06370 ; SYS3, 4E80, 501B
06380 ;
06390 ;
06400 ;
06410 ;*** 013 ****
06420 ;
06430 ; 'LOAD' IN BASIC SETS 'END+1 OF TEXT' IN 40F9, A ONE BYTE TOO HIGH. TENDS TO
06440 ; MAKE BASIC PROGRAM MODULES 1 BYTE TOO LONG, BUT HAS NO NOTICEABLE ERRONEOUS
06450 ; EFFECT.
06460 ; BASIC, 5FDC
06470 ;
06480 ;
06490 ;
06500 ;*** 014 ****
06510 ;
06520 ; DOS SYS0, INIT INITIALIZES 4158-43FF AND THEN, THROUGHOUT SUBSEQUENT SYSTEM
06530 ; OPERATION, NEVER USES IT, EXCEPT FOR UNEXPLAINED REFERENCE TO 4359 IN SYS1
06540 ; A REFERENCE TO 43C0-FF BY SYS6 FOR 'DEVICE', AND AN UNINTENTIONAL REFERENCE
06550 ; TO 43FFH AS A RESULT OF INSTRUCTIONS IN SYS0, 46C1-CB.
06560 ; I HAVE CRITICAL NEED FOR A DOS RESIDENT PATCH AREA.
06570 ; I WILL GAMBLE THAT THIS AREA IS TRULY AVAILABLE AND USE IT.
06580 ; 'DEVICE' COMMAND IS DISABLED IN THIS SYSTEM.
06590 ;***** EXTREME CAUTION!!!!!!!!!! *****
06600 ; SYS0, 4E20
06610 ; SYS6, 531D
06620 ;***** SEE SUBSEQUENT PATCHES FOR THIS AREA'S USE *****
06630 ;
06640 ;
06650 ;
06660 ;*** 015 ****
06670 ;
06680 ; DOS MISUSES, AT TIMES, THE CONCEPT OF EOF (FIDE+15H, 14H, 03H) (FCB+00H, 0CH, 08H)
06690 ; AND OF CURRENT POSITION WITHIN THE FILE (FCB+0BH, 0AH, 05H).

06700 ;THESE TWO 3 BYTE VALUES REPRESENT RELATIVE BYTE POSITIONS WITHIN THE FILE OF:
06710 ; 1. THE 1ST BYTE AFTER THE LOGICAL END OF FILE.
06720 ; 2. THE NEXT BYTE TO BE PROCESSED EITHER BY READ OR WRITE.
06730 ;THESE CHANGES ARE TO:
06740 ; 1. ALLOW PROPER OPERATION UNDER THIS CONCEPT.
06750 ; 2. YET, PRESERVE EOF VALUE IN FPDE IN DOS ERRONEOUS STATE SO FILES
06760 ; CAN BE USED IN EITHER THE 'ORIGINAL' DOS SYSTEM OR MY 'ALTERED'
06770 ; SYSTEM, BUT NOT A MIXTURE OF BOTH.
06780 ;SYS2, 4FE8
06790 ;SYS3, 4E19
06800 ;SYS0, 4864, 47F6, 47C5, 47CC
06810 ;
06820 ;
06830 ;
06840 ;*** 016 ****
06850 ;
06860 ;IF THE NEW KEY PRESSED IS THE SAME AS THE LAST KEY ACCEPTED BY THE SYSTEM, THEN
06870 ;REQUIRE THAT IT STILL BE PRESSED UPON EXIT FROM THE DECODE ROUTINE, WHICH
06880 ;INCLUDES A DELAY OF APPROXIMATELY 50 MS.
06890 ;THIS IS AN ATTEMPT TO REDUCE THE NUMBER OF UNINTENTIONAL REPEATED KEYSTROKES.
06900 ;THIS CHANGE INSERTS 'ZAP07' ADDR (IN DOS RESIDENT ROUTINE SYS0) INTO
06910 ;4016-7 (ROUTINE ADDR IN KEYBOARD CONTROL BLOCK) AND CALLS 03FAH, VICE 03E3H,
06920 ; TO GET THE CURRENT KEY. THIS MIGHT MEAN TROUBLE FOR ANY OF THE FOLLOWING:
06930 ; 1. OTHER USERS WHO CHANGE THIS ADDRESS VALUE IN THE CONTROL BLOCK.
06940 ; 2. ROUTINES WHICH LOAD DIRECTLY OR INDIRECTLY INTO THE DOS AREA,
06950 ; 4000-51FF.
06960 ;*** NOTE: THIS 'DEBOUNCE ROUTINE' WILL NOT ACTIVATE (LEAVING THE ADDR AT
06970 ;4016-7 UNCHANGED) IF SHIFT-UP-ARROW IS PRESSED FROM RESET OR POWER-ON
06980 ; UNTIL THE 1ST MSG APPEARS ON THE DISPLAY.
06990 ; WITH THE DEBOUNCE ROUTINE DEACTIVATED, 'JKL' OPERATION IS ALTERED
07000 ;SOMEWHAT. (SEE CHANGE #017).
07010 ; SYS0, 4E20
07020 ;
07030 ;
07040 ;
07050 ;*** 017 ****
07060 ;
07070 ;IN CONJUNCTION WITH CHANGE #016, THIS CHANGE ALLOWS THE FULL DISPLAY CONTENTS
07080 ;TO BE SENT AS SIXTEEN 64 CHAR LINES TO THE LINE PRINTER WHENEVER THE 3 KEYS
07090 ; J, K AND L ARE SIMULTANEOUSLY DEPRESSED AND ONE OF THE FOLLOWING EXISTS:
07100 ; 1. 'DEBOUNCE' IS ACTIVATED (SEE CHANGE #017), AND A USER PROGRAM
07110 ; IS TESTING FOR A DEPRESSED KEY. (J, K OR L IS NOT
07120 ; SEEN BY THE CALLING PROGRAM).
07130 ; 2. CLOCK INTERRUPTS ARE ENABLED. IF 'DEBOUNCE' IS NOT ACTIVATED,
07140 ; ONE OR MORE OF J, K OR L MAY BE INPUTTED TO A PROGRAM
07150 ; WAITING ON KEYBOARD INPUT AT THIS TIME.
07160 ;THE ONLY EDITING DONE BY THIS ROUTINE ENROUTE TO THE PRINTER IS TO CHANGE
07170 ;ALL CHARS WHOSE HEXIDECIMAL VALUE >= 80H TO PERIODS. THIS WILL CAUSE
07180 ;GRAPHICS TO PRINT AS PERIODS.
07190 ;
07200 ;SYS0, 4520, 4E20
07210 ;
07220 ;
07230 ;
07240 ;*** 018 ****
07250 ;
07260 ;L2 BASIC DOES NOT FULLY CONVERT BOTH SHIFT AND NON-SHIFT VALUES FOR THE LETTERS
07270 ;A-Z AND THE SYMBOL @ TO ASCII UPPER CASE. (BASIC'S KEY INPUT ROUTINE ACTUALLY
07280 ; CONVERTS THESE CHARS INITIALLY FROM LOWER TO UPPER AND UPPER TO LOWER. THEN
07290 ; THE SOURCE PACK ROUTINE (1BC0) CONVERTS MOST BUT NOT ALL OF THE NOW LOWER

07300 ;(ORIGINALLY UPPER) CASE CHARS BACK TO UPPER CASE.)
07310 ;PROPER BASIC PROGRAM OPERATION REQUIRES ALL CHARACTERS IN THE TEXT TO BE UPPER
07320 ;CASE FOR @ AND A-Z. ERRORS IN THIS REGARD ARE NOW CORRECTED WHEN LINE IS
07330 ;ENTERED (OR EDITED). LOWER-CASE TEXT IS RETAINED BY NEWDOS IF SUCH LINES ARE
07340 ;ENTERED AND EDITED BY TRSDOS ONLY.
07350 ;
07360 ;THIS 'FILTER' MAY BE INHIBITED USING SUPERZAP; ON NEWDOS+ DISKETTE TR.3, SEC.5,
07370 ;MOD 6D & 6E FROM 00 52 TO 33 60.
07380 ;
07390 ; BASIC, 5523
07400 ;
07410 ;
07420 ;*** 019 *****
07430 ;
07440 ;PROVIDE DOS-BASIC WITH AN OPEN "E" FUNCTION WHICH:
07450 ; 1. OPENS A FILE
07460 ; 2. SETS 'NEXT' INDICATOR = 'EOF', (FCB+5,A,B := FCB+8,C,D)
07470 ; 3. SETS PROCESSING TO CONTINUE AS IF OPEN "O".
07480 ;THIS ALLOWS THE USER TO ADD TO A SEQUENTIAL FILE.
07490 ;IF THE FILE WAS CREATED BY OPEN OR WAS NOT YET PREVIOUSLY WRITTEN TO
07500 ; (IE, EOF = 0), THEN OPEN "E" OPERATES EXACTLY AS OPEN "O".
07510 ; BASIC, 6372, 6386
07520 ;
07530 ;
07540 ;*** 020 *****
07550 ;
07560 ;'EOF' FUNCTION WILL MALFUNCTION UNDER THE REMOTE POSSIBILITY OF 'NEXT' > 'EOF'
07570 ;AND THE EOF BYTE AT MISMATCH = 0
07580 ; BASIC, 61BF, 6200
07590 ;
07600 ;
07610 ;*** 021 *****
07620 ;
07630 ;'LOC' COMMAND IN DOS-BASIC FAILS DUE TO ERRONEOUS CALL.
07640 ; BASIC 6231
07650 ;
07660 ;
07670 ;*** 022 *****
07680 ;
07690 ;DOS-BASIC'S FILESPEC EDIT ROUTINE ALLOWS ONLY A 22 CHAR FILESPEC WHEREAS
07700 ;SYSTEM DEFINITION ALLOWS FOR 23 (NNNNNNNN/TTT.PPPPPPP:D)
07710 ; BASIC, 630C
07720 ;
07730 ;
07740 ;*** 023 *****
07750 ;
07760 ;'APPEND' DOS COMMAND JUST DOESN'T WORK. HOPEFULLY NOW IT DOES.
07770 ; SYS6, 573D
07780 ;
07790 ;
07800 ;*** 024 *****
07810 ;
07820 ;WHEN DOING LOGICAL TRANSFER (BYTE OR LOGICAL RECORD) (BIT 7,FCB+1 ON), THE
07830 ;OUTPUT SECTOR BUFFER MUST BE DIFFERENT FROM THE INPUT SECTOR BUFFER.
07840 ; SYS6, 5720
07850 ;
07860 ;
07870 ;*** 025 *****
07880 ;
07890 ;'VERIFY' CHANGES THE WRONG INSTRUCTION IN SYS0, CHANGING THE WRITE-WITH-VERIFY

```

07900 #VECTOR RATHER THAN THE WRITE-WITHOUT-VERIFY VECTOR.
07910 # SYS6: 550F
07920 #
07930 #
07940 #
07950 #*** 026 *****
07960 #
07970 #'ATTRIB' USES THE WRONG ERROR CODE
07980 # SYS6: 5821
07990 #
08000 #
08010 #
08020 #*** 027 *****
08030 #
08040 #'ATTRIB' BAD BRANCH
08050 # SYS6: 5B45
08060 #
08070 #
08080 #
08090 #*** 028 *****
08100 #
08110 #PROVIDE 'ATTRIB' WITH CAPABILITY OF TURNING OFF THE INVISIBLE ATTRIBUTE.
08120 # SYS6: 5C02
08130 #
08140 #
08150 #
08160 #*** 029 *****
08170 #
08180 # SYS3 CHANGED TO CAUSE 'KILL' TO ZERO 1ST BYTE OF RELEASED FPDE OR FXDE, RATHER
08190 # THAN JUST RESETTING THE ENTRY-IN-USE BIT. THIS AGREES WITH 'CLOSE'.
08200 # SYS3: 4F00
08210 #
08220 #         END SYSMODS2/CON; APPEND SYSMODS3/CON NEXT;
08230 #         SYSMODS3/CON
08240 #
08250 #*** 030 *****
08260 #
08270 #'CLOSE' IN SYS3 CAUSES A MAJOR SYSTEM DISASTER WHEN IT RELEASES A FXDE BY NOT
08280 #PRESERVING THE CONTENTS OF REC DL, CONTAINING COUNT 11 OF SECTORS YET TO BE FREED,
08290 #WHEN FREEING A NO LONGER NEEDED FXDE.
08300 #THIS ERROR IS COMPOUNDED BY THE BRANCH AT 4ED9 NOT IMPLICITLY ENDING DEALLOCATION
08310 #WHEN THE FILE IS KNOWN TO HAVE NO MORE GRANULES ASSIGNED.
08320 #
08330 #THESE ERRORS CAUSE ALL WRITEABLE MAIN MEMORY FROM 3000 - 42XX TO BE SET = 'FF',
08340 #WHERE XX IS RELATIVE POSITION WITHIN THE SECTOR OF THE LAST BYTE OF THE FPDE
08350 #POINTED TO BY THE LAST FXDE RELEASED.
08360 #THE CORRESPONDING SECTOR IN THE DIRECTORY IS ALSO FILLED WITH 'FF' TO THAT
08370 #RELATIVE POINT.
08380 #FURTHER, AS THIS GOES ON, THE CAT DIRECTORY SECTOR IS MODIFIED TO FREE UP
08390 #GRANULES AT RANDOM IN TRACKS 00-FF, WITH MOST TRACKS BELOW 80H.
08400 # UNDETECTED, THIS WILL CAUSE GRANULES PREVIOUSLY ALLOCATED TO OTHER FILES TO
08410 #BE ALLOCATED AGAIN IN SUBSEQUENT FILE ALLOCATIONS.
08420 #THIS INCLUDES REALLOCATION OF BOOT/SYS AND DIR/SYS GRANULES, EVENTUALLY CLOBBERING
08430 #THEM.
08440 #ALSO, FILES WHOSE FPDE PRECEDED THE DESTROYED FPDE IN THE DIRECTORY ENTRY SECTOR
08450 #WILL DISAPPEAR FROM THE SYSTEM AND IF A FILE'S FXDE WAS SO DESTROYED, THE USER
08460 #SHOULD HAVE HORRENDOUS TROUBLE FROM THAT FILE, CONSIDERED LUCKY IF THE SYSTEM
08470 #DETECTS AN ERROR
08480 #
08490 #***** ALMOST AS BAD; HL IS NOT DECREMENTED TO 2ND BYTE OF NEXT LOWER

```


08500 #EXTENT NOR IS IT PROTECTED BY THE DIRECTORY SECTOR WRITE CALL AT 4F08.
08510 #THIS CAUSES THE TWO BYTES(WHATEVER THEY ARE) AT 41FF-4200 TO BE USED AS THE
08520 #NEXT LOWER EXTENT FOR THE FILE, CAUSING A SOMEWHAT RANDOM DEALLOCATION OF
08530 #GRANULES, USUALLY IN THE RANGE OF TRACKS 00-10H.
08540 #
08550 #YET MORE. IF A NEW FXDE IS ALLOCATED TO THE FILE AND THEN THE DISKETTE
08560 #IS FOUND TO BE FULL, SYS3 MALFUNCTIONS (AT SOME FUTURE TIME) WHEN CLOSE TRIES
08570 #TO FREE THE SPACE ASSIGNED TO THAT FXDE, ASSUMING THERE IS SOME WHEN THERE IS
08580 #NONE.
08590 #
08600 # SYS3, 4ED9, 4FFD, 500C, 4F08, 4E92, 4ED1
08610 #
08620 #
08630 #
08640 ##### 031 #####
08650 #
08660 #CHANGE TO DOS-BASIC TO ALLOW THE FOLLOWING 'DIRECT' COMMANDS:
08670 # 1. '.' (PERIOD), LIST CURRENT TEXT LINE.
08680 # 2. 'DOWN-ARROW', LIST NEXT TEXT LINE. IF NO NEXT TEXT LINE, PERFORMS
08690 # AS 'SHIFT-DOWN-ARROW'
08700 # 3. 'UP-ARROW', LIST TEXT LINE BEFORE CURRENT LINE. IF NONE, PERFORMS
08710 # AS 'SHIFT-UP-ARROW'.
08720 # 4. 'SHIFT-UP-ARROW', LIST 1ST TEXT LINE.
08730 # 5. 'SHIFT-DOWN-ARROW', LIST LAST LINE IN TEXT.
08740 # 6. ',' (COMMA), 'EDIT' CURRENT TEXT LINE.
08750 #
08760 #ONLY 1 SUCH COMMAND PER DIRECT STATEMENT LINE, AND THE COMMAND, TO BE SEEN,
08770 #MUST BE THE FIRST CHAR OF THE INPUT LINE (NO SEQUENCE NUMBER ALLOWED AND
08780 #BACKSPACING NOT ALLOWED).
08790 # BASIC, 5993
08800 #
08810 #
08820 #
08830 ##### 032 #####
08840 #
08850 #IN 'FORMAT', ALLOW THE USER THE OPTION OF CONTINUING THE FORMAT WHEN DISKETTE
08860 #FOUND TO CONTAIN DATA.
08870 # FORMAT, 542C
08880 #
08890 #
08900 #
08910 ##### 033 #####
08920 #
08930 #NEED TO POWER UP DRIVE AGAIN IF ADDITIONAL INPUT WAITS PATCHED IN.
08940 # FORMAT, 5464
08950 #
08960 #
08970 #
08980 ##### 034 #####
08990 #
09000 #'FORMAT' DOES NOT DAMAGE DOS AREAS, THEREFORE, A SYSTEM RESET IS NOT
09010 #NECESSARY. UPON FINAL 'ENTER', EXIT VIA 4020H IF FORMAT COMPLETED AND VIA
09020 #4030H IF FORMAT DID NOT FULLY COMPLETE.
09030 # FORMAT, 56AA
09040 #
09050 #
09060 #
09070 ##### 035 #####
09080 #
09090 # CHANGES TO ALLOW 'DOS' COMMANDS TO BE EXECUTED WITHIN BASIC'S 'CMD' COMMAND.

```

09100 ;THE 'CMD' FUNCTION IS USED, PREEMPTING THE SITUATION WHERE THE QUOTE
09110 ;STRING CONTAINS MORE THAN ONE CHAR. IF THE LAST CHAR OF THE QUOTE STRING
09120 ;IS A '$', THEN THE STRING IS CONSIDERED TO BE A CHARACTER STRING VARIABLE
09130 ;NAME, AND THE CONTENTS OF THAT VARIABLE'S STRING IS USED AS THE QUOTE STRING
09140 ;FOR THE 'CMD' FUNCTION AS THE DOS COMMAND.
09150 ; THIS PATCH USES 444B-4466, AN AREA ASSUMED AVAILABLE IN SYS0.
09160 ;
09170 ; CMD"XX" WILL CAUSE 'OM' (OUT OF MEMORY) ERROR TO OCCUR IF THERE IS
09180 ;INSUFFICIENT FREE SPACE BETWEEN BOTTOM OF STRING AREA AND TOP OF ARRAY AREA.
09190 ; CURRENTLY, ABOUT 8K OF FREE SPACE IS REQUIRED. BEFORE GOING TO DOS,
09200 ;A CHECKSUM IS COMPUTED OVER THE ENTIRE MAIN MEMORY AREA OF BASIC THAT
09210 ;IS NOT ALLOWED CHANGED BY THE DOS FUNCTION. THIS TAKES ABOUT 2 SECONDS.
09220 ;IF THE DOS FUNCTION RETURNS TO BASIC, THIS CHECKSUM IS COMPUTED AGAIN
09230 ;(ANOTHER 2 SECS). IF THE CHECKSUM IS BAD, BASIC CANNOT CONTINUE AND AN
09240 ;EXIT TO 'DOS READY' IS TAKEN.
09250 ;
09260 ;THE PROGRAM OR FUNCTION INVOKED BY CMD"XX" MAY RETURN TO BASIC, TO CONTINUE
09270 ; THE BASIC PROGRAM'S EXECUTION, IF IT DID NOT USE ANY OF MEMORY BETWEEN 6100
09280 ;AND BASIC'S HIGH ADDR (MEM SIZE) AND IT TAKES ONE OF THE FOLLOWING EXITS:
09290 ;      1. 402D    SUCCESSFUL COMPLETION EXIT. NEXT BASIC STATEMENT WILL BE
09300 ;                  EXECUTED.
09310 ;      2. 4030    UNSUCCESSFUL COMPLETION EXIT. COMPLAINT HAS ALREADY
09320 ;                  BEEN DISPLAYED. IF CMD"D" NOT ACTIVE, 'UNPRINTABLE ERROR'
09330 ;                  (DECIMAL 38) BASIC ERROR WILL BE INVOKED,
09340 ;                  CAUSING EITHER THE 'ON ERROR' ROUTINE TO BE INVOKED OR
09350 ;                  PROGRAM EXECUTION TERMINATED IN ERROR.
09360 ;                  IF CMD"D" IS ACTIVATED, 4030 EXIT GOES TO DEBUG DISPLAY,
09370 ;                  FROM WHICH 'RESET' MUST BE ISSUED WHEN DONE WITH DEBUG.
09380 ;      3. 4409    UNSUCCESSFUL COMPLETION EXIT. DEPENDING UPON THE SYSTEM
09390 ;                  ERROR CODE, A BASIC ERROR CODE WILL BE INVOKED, CAUSING 1
09400 ;                  OF THE TWO (CMD"D" INACTIVE) OPTIONS IN THE 4030 EXIT.
09410 ;                  CMD"E" CAN DISPLAY THE ACTUAL SYSTEM ERROR MSG.
09420 ;
09430 ;***** CAUTION. IF THE INVOKED DOS FUNCTION (IE, A SINGLE DRIVE-TWO
09440 ;DISKETTE COPY OR OTHER OBJECT CODE PROGRAMS SUCH AS 'EDITASM')
09450 ;USES BASIC'S MEMORY ABOVE 6F00, EXITING THROUGH ONE OF THE
09460 ;ABOVE THREE ADDRESSES WILL CAUSE UNPREDICTABLE RESULTS AS BASIC'S CODE IS
09470 ;CLOBBERED. WITH LUCK, RESET WILL TAKE PLACE.
09480 ;
09490 ; BASIC, 56CD
09500 ;SYS0 444B-4466
09510 ;
09520 ;
09530 ;
09540 ;*** 036 *****
09550 ;
09560 ;CHANGES TO SPEED UP 'BASIC'S 'LOAD' AND 'SAVE' OPERATIONS BY USING SECTOR
09570 ; I/O TO DOS SYSTEM RATHER THAN BYTE I/O.
09580 ; FOR LARGE BASIC PROGRAMS, LOAD AND SAVE RUN AT APPROX 33% AND 50% OF
09590 ;PREVIOUS TIMES RESPECTIVELY.
09600 ;
09610 ;BASIC, 5FB6, 6055
09620 ;
09630 ;
09640 ;
09650 ;*** 037 *****
09660 ;
09670 ;CHANGES TO 'FORMAT' TO ALLOW IT TO EXECUTE WITH PARAMETERS SPECIFIED ALONG
09680 ;WITH THE FORMAT COMMAND.
09690 ;

```

```

09700 ;      1. FORMAT D,NNNNNNNN,MM/DD/YY,PPPPPPPP
09710 ;      2. FORMAT D,NNNNNNNN,MM/DD/YY,PPPPPPPP,Y
09720 ;
09730 ;WHERE
09740 ;      D = RELATIVE NUMBER OF DRIVE TO BE FORMATTED
09750 ;      NNNNNNNN = 1-8 CHAR NAME FOR DISKETTE
09760 ;      MM/DD/YY = CREATION DATE
09770 ;      PPPPPPPP = 1-8 CHAR PASSWORD
09780 ;      Y PARAM IS USED IF OK THAT DISKETTE BEING FORMATTED CONTAINS DATA.
09790 ;
09800 ; IF 'D' IS DRIVE 0 (THE SYSTEM DRIVE); FORMAT WILL ASK FOR THE
09810 ; PROPER FORMAT/DESTINATION OR SOURCE DISKETTE TO BE MOUNTED; AND UPON EXIT WILL
09820 ; ASK FOR A SYSTEM DISKETTE TO BE MOUNTED BACK ONTO DRIVE 0.
09830 ;
09840 ; 'FORMAT' IS ALSO MODIFIED SUCH THAT IF COMMAND PARAMS ARE SUPPLIED WITH THE
09850 ; COMMAND, THE COMMAND CAN BE RETRIED AFTER USER HAS CORRECTED THE PROBLEM;
09860 ; PROVIDED THE COMMAND PARAMS THEMSELVES ARE NOT IN ERROR.
09870 ;
09880 ; FORMAT, 522E, 5278, 52A0, 52AE, 52B6, 52BE, 52CF, 5341
09890 ;
09900 ;
09910 ;
09920 ; *** 038 *****
09930 ;
09940 ; FORMAT IS ALTERED TO USE 5000 AS BASE OF FULL TRACK BUFFER; VICE 5F00.
09950 ; THIS DECREASES FORMAT EXECUTION AREA SIZE TO REDUCE SIZE SAVED BY BASIC WHEN
09960 ; EXECUTING DOS COMMANDS.
09970 ; FORMAT 5499, 5526, 5551
09980 ;
09990 ;
10000 ;
10010 ; *** 039 *****
10020 ;
10030 ; 'BASIC IS MODIFIED SUCH THAT ITS ACTIVATION SEQUENCE IS ONE OF THE FOLLOWING:
10040 ;      1. BASIC
10050 ;      2. BASIC *
10060 ;      3. BASIC N
10070 ;      4. BASIC M
10080 ;      5. BASIC CMD
10090 ;      6. BASIC N;M;CMD
10100 ;      7. BASIC M;N;CMD
10110 ;      8. BASIC N;N
10120 ;      9. BASIC M;N
10130 ;     10. BASIC N;CMD
10140 ;     11. BASIC M;CMD
10150 ;
10160 ;WHERE:
10170 ;      * MEANS THE USER WANTS BASIC TO REINSTITUTE THE PROGRAM IN THE TEXT
10180 ;      BUFFER, USING THE SAME VALUES FOR M AND N AS APPEAR TO EXIST IN
10190 ;      MAIN MEMORY. THIS ALLOWS THE USER TO RECOVER FROM AN UNWANTED
10200 ;      'RESET' OR TO GET BACK TO THE SAME PROGRAM AFTER A CMD'S".
10210 ;      IF BASIC IS ABLE TO ACCOMPLISH THIS, IT FORCES 'LIST' AS ITS
10220 ;      1ST COMMAND.
10230 ;
10240 ;      N = # OF I/O AREAS. DEFAULT : 3. MAX = 15.
10250 ;      M = MEMORY SIZE. DEFAULT IS ALL THAT THE COMPUTER HAS.
10260 ;      CMD = ANY BASIC DIRECT COMMAND. THIS COMMAND WILL BE EXECUTED AS
10270 ;      SOON AS INITIALIZATION IS COMPLETED.
10280 ;
10290 ; ANY ERROR ENCOUNTERED DURING INITIALIZATION CAUSES A RETURN TO DOS.

```

```

10300 ;THE OLD BASIC STARTUP PROCEDURE IS HEREBY SUPERSEDED.
10310 ;
10320 ;BASIC, 5200, 5BB0, 5BF0, 5C35, 5C54, 5CA4, 5C65, 5C88
10330 ;
10340 ;
10350 ;*** 040 ****
10360 ;
10370 ; 'COPY' IS ALTERED TO HANDLE THE FORMATS:
10380 ;     1. COPY FILESPEC1 TO FILESPEC2 (SAME AS OLD COPY)
10390 ;     2. COPY $FILESPEC1 TO FILESPEC2
10400 ;     3. COPY :D FILESPEC1 TO FILESPEC2
10410 ;     4. COPY :D $FILESPEC1 TO FILESPEC2
10420 ;     5. COPY :D TO :E MM/DD/YY
10430 ;WHERE
10440 ;     1. D AND E ARE DRIVE #'S AND MAY BE :.
10450 ;     2. FILESPEC1 IS THE 'SOURCE' FILE'S FILESPEC.
10460 ;     3. FILESPEC2 IS THE 'DESTINATION' FILE'S FILESPEC.
10470 ;     4. $ MEANS THAT EITHER THE SOURCE OR THE DESTINATION FILE OR BOTH
10480 ;        ARE ON A DISKETTE CONTAINING AN ALIEN SYSTEM (IE, NOT THIS SYSTEM)
10490 ;        IF D IS SPECIFIED AND IS NOT : 0, THEN $ SHOULD NOT BE SPEC'ED
10500 ;        AS THE NECESSARY COMPATIBLE SYSTEM ROUTINES, SYS2 AND SYS3, WILL
10510 ;        BE TAKEN FROM DRIVE 0, WHICH IS ASSUMED TO CONTAIN 'THIS' SYSTEM.
10520 ;     5. FORMATS 3 AND 4 INDICATE THAT SOURCE AND DESTINATION FILES ARE
10530 ;        ARE ON DIFFERENT DISKETTES, BUT THAT ONLY THE SPECIFIED DRIVE WILL
10540 ;        BE USED TO CONTAIN THOSE DISKETTES.
10550 ;     6. FORMAT 5 PERFORMS A BACKUP FUNCTION. BEFORE DATA IS
10560 ;        WRITTEN ON THE DESTINATION DISKETTE, IT IS FORMATTED TO ALL DATA
10570 ;        SECTORS (NO BOOT OR DIRECTORY). THE ENTIRE CONTENTS OF THE SOURCE
10580 ;        DISKETTE ARE THEN COPIED ONTO THE DESTINATION DISKETTE (INCLUDING
10590 ;        BOOT AND DIRECTORY), AND THE 'MM/DD/YY' VALUE INSERTED AS THE
10600 ;        CREATION DATE FOR THE DESTINATION DISKETTE.
10610 ;     7. WHEN TWO DISKETTES, BUT ONLY 1 DRIVE, ARE SPECIFIED FOR A COPY,
10620 ;        ALL OF MEMORY ABOVE COPY'S CODE IS USED AS BUFFERS. ALL OTHER
10630 ;        CASES USE A ONE TRACK BUFFER, THUS ALLOWING RETURN TO BASIC, IF
10640 ;        COPY WAS INVOKED VIA BASIC'S CMD"XX" STATEMENT.
10650 ;     8. WHENEVER A SPACE IS USED AS A SEPARATOR, ONE AND ONLY ONE IS ALLOWED.
10660 ;
10670 ; WHEN THE LEADING PORTIONS OF FILESPEC2 ARE THE SAME AS THAT OF FILESPEC1,
10680 ;IT MAY BE LEFT OFF FILESPEC2 WITH FILESPEC2 STARTING WITH ONE OF / OR . OR : 0.
10690 ;INDICATE THE DIFFERING PORTION.
10700 ; EXAMPLE:
10710 ;     COPY ABCD/TIT:0 TO ABCD/TIT:1
10720 ;MAY BE WRITTEN AS
10730 ;     COPY ABCD/TIT:0 TO :1
10740 ;
10750 ; 'COPY' IS NO LONGER EXECUTED AS A FUNCTION WITHIN SYS6. SYS1 IS MODIFIED
10760 ;TO ROUTE 'COPY' TO COPY/CMD MODULE. SYS6 IS MODIFIED TO NOT USE ITS 'COPY'
10770 ;FUNCTION. DUE TO ATTEMPTS TO RESTRICT COPY/CMD TO 1 GRANULE, THE FORMAT
10780 ;PORTION OF COPY HAS BEEN PLACED IN SYS3, INVOKED BY THE
10790 ;70 CODE.
10800 ;
10810 ;ALSO, OLD 'COPY' MALFUNCTIONED IF LAST SECTOR BEFORE EOF SECTOR IS NOT FULL
10820 ; (IE, FPDE+3 AND FCB+8 <> 0) BECAUSE IT SECTOR READ/WITES RATHER THAN
10830 ;SINGLE BYTE READ/WITES. THIS CHANGE CORRECTS THIS.
10840 ;
10850 ;COPY/CMD
10860 ;SYS1 4EED
10870 ;SYS6 5216
10880 ;SYS3 ZAP15, FORMAT
10890 ;

```

10900 ;
10910 ;
10920 ;*** 041 ***
10930 ;
10940 ;PROGRAM LOADER IN SYS0 IS ALTERED TO USE 4200-42FF, VICE 4D00-4DFF AS IT'S
10950 ;I/O BUFFER, THUS ALLOWING SYSTEM PROGRAMS TO LOAD INTO 4D00-4DFF, IF DESIRABLE.
10960 ;A CHECK IS MADE IN THE ACTUAL I/O CODE TO DISABLE REMEMBRANCE OF THAT
10970 ;SYSTEM PROGRAM BEING IN MAIN MEMORY IF ITS ENTRY POINT WAS WITHIN 4D00-4DFF
10980 ; AND A READ/WRITE SECTOR TAKES PLACE ON THAT AREA.
10990 ;SYS0 4C18, 44A4, 4C39
11000 ;
11010 ;
11020 ;
11030 ;*** 042 ***
11040 ;
11050 ;IF A SYSTEM MODULE BEING LOADED USING THE 16 BYTE FCB (HALF SIZE) AT 44A0,
11060 ;AN 'RECORD OUT OF RANGE' (1DH) ERROR IS CALLED IF FILE HAS MORE THAN
11070 ;ONE EXTENT.
11080 ;SYS0 4912
11090 ;
11100 ;
11110 ;
11120 ;*** 043 ***
11130 ;
11140 ;IF THE SYSTEM MODULE 'CALL' ROUTINE FINDS THE MODULE'S DIRECTORY ENTRY
11150 ;INACTIVE, THEN ONE OF THE FOLLOWING:
11160 ; 1. IF SYS4 IS AN ACTIVE MODULE IN THE SYSTEM, THEN 'FILE NOT IN DIRECTORY'
11170 ; ERROR WILL BE DISPLAYED VIA JUMP TO SYS4 AT 4409H (IF CMD"XX",
11180 ; BASIC WILL INTERCEPT).
11190 ; 2. IF THE JUMP TO SYS4 VIA 4409 FINDS SYS4 NOT IN THE SYSTEM, THEN
11200 ; 'RESET' WILL BE INVOKED.
11210 ;SYS0 4BD4
11220 ;
11230 ;
11240 ;
11250 ;*** 044 ***
11260 ;
11270 ;TO REDUCE THE RESIDENT SIZE OF BASIC (DUE TO ADDED CODE) THE FOLLOWING
11280 ;STEPS HAVE BEEN TAKEN:
11290 ; 1. SYS13/SYS MODULE CREATED TO CONTAIN DOS BASIC'S
11300 ; ERROR DISPLAY CODE AND TABLE.
11310 ; 2. 5200-5555: ENTRY TO 5200 IS CHANGED TO ZAP18, ERROR MESSAGES
11320 ; AND ERROR DISPLAY CODE FROM 57B2-57E0 ARE MOVED TO SYS13.
11330 ; THE JUMP VECTORS (54C3-5555) ARE MOVED TO 4D00 AS THEY ARE NEEDED
11340 ; ONLY DURING INITIALIZATION.
11350 ; 3. 57B2-5823: THE ERROR DISPLAY CODE (57B2-57E0) IS MOVED TO SYS13.
11360 ; CODE (57E1-573F) IS DELETED AND CMD"X" DISABLED.
11370 ; THE AUTHORED/COPYRIGHTED MSG IS APPENDED, DISPLAYED DURING
11380 ; INITIALIZATION AND THEN ITS STORAGE REUSED.
11390 ; 4. 5BA4-5BC7: INITIALIZATION CODE GREATLY MODIFIED. COMMAND BUFFER
11400 ; SHIFTED TO 5BAD-5C9D TO OVERLAP INITIALIZATION CODE, WHICH IS
11410 ; NOT NEEDED AFTER INITIALIZATION. THE ROUTINE AT 5C8B-5C90
11420 ; NOW AT 5BA4-5BA9.
11430 ; 5. 5A15-5B13: ROUTINE AT 5A5B-5A66 MOVED TO 5A15-5120, HAVE NO IDEA
11440 ; WHAT THE REST OF THE AREA WAS USED FOR; SO WILL USE IT FOR
11450 ; APPENDAGE CODE.
11460 ; 6. IF NECESSARY, MORE APPENDAGE CODE CAN BE ADDED AT 6431.
11470 ;
11480 ;
11490 ;

```

11500 *** 045 *****
11510 #
11520 # THROUGH NEW SYSTEM MODULE 'SYS12/SYS' THE FOLLOWING BASIC COMMAND IS ALLOWED:
11530 #
11540 #     1. REF%      DISPLAY FULL REFERENCE LIST.
11550 #     2. REF$      PRINT FULL REFERENCE LIST ON LINE PRINTER.
11560 #     3. REFNN     DISPLAY ALL REFERENCES TO THE VARIABLE(S) NAMED NN.
11570 #                  IF NN IS ONLY 1 CHAR, BLANK IS ASSUMED FOR THE SECOND.
11580 #                  NN MAY NOT BE MORE THAN 2 CHARS AND MUST NOT HAVE A TYPE SUFFIX.
11590 #     4. REFSSSSS  DISPLAY ALL REFERENCES TO THE INTEGER SSSSS WHERE
11600 #                  SSSSS IS A 1-5 DECIMAL DIGIT # BETWEEN 0 AND 99999.
11610 #                  HEXIDECIMAL OR OCTAL REFERENCES WITHIN THE TEXT ARE NOT LISTED.
11620 #     5. REF*NN
11630 #     6. REF$NN
11640 #     7. REF*SSSSS
11650 #     8. REF$SSSSS
11660 #     9. REF       DISPLAY THE NEXT TEXT LINE CONTAINING AT LEAST ONE
11670 #                  REFERENCE TO THE VARIABLE OR NUMBER SPECIFIED BY THE LAST
11680 #                  REFNN OR REFSSSSS COMMAND EXECUTED. IF NO MORE REFERENCING
11690 #                  TEXT LINES, 'TEXT END' WILL BE DISPLAYED.
11700 #                  IF 'REF' ENTERED AGAIN, THE FIRST REFERENCING TEXT LINE
11710 #                  WILL BE LISTED.
11720 #                  REMEMBRANCE OF THE LAST VARIABLE NAME, NUMBER OR POSITION
11730 #                  WITHIN THE TEXT IS USUALLY (BUT NOT ALWAYS) LOST WHEN ANY
11740 #                  COMMAND INVOLVING DOS IS EXECUTED.
11750 #
11760 #                  PRESS BREAK TO PAUSE, ENTER TO CONTINUE.
11770 # FORMATS 5-8 ARE THE SAME AS 1 AND 2, EXCEPT LISTING/PRINTING STARTS WITH
11780 # THE SPECIFIED VARIABLE NAME OR DECIMAL NUMBER, IF IT EXISTS, OR THE NEXT
11790 # HIGHER EXISTING NAME OR NUMBER, IF NOT.
11800 #
11810 # IF 'SYS12/SYS' NOT IN THE SYSTEM WHEN COMMAND 'REF' EXECUTED, BASIC ERROR
11820 # 'UNPRINTABLE ERROR'(CODE 38 DECIMAL) WILL BE INVOKED. CMD"E" WILL DISPLAY
11830 # 'FILE NOT IN DIRECTORY'
11840 #
11850 #
11860 *** 046 *****
11870 #
11880 # NEW SYSTEM MODULE 'SYS11/SYS' ALLOWS THE FOLLOWING BASIC COMMAND:
11890 #
11900 #     1. RENUM U    SEARCHES TEXT FOR UNDEFINED SEQUENCE NUMBERS AND
11910 #                  FOR SOME ERRORS ASSOCIATED WITH BASIC STATEMENTS THAT USE
11920 #                  SEQUENCE NUMBERS. ERRORS ARE INDICATED:
11930 #                  SSSSS/U - THERE IS NO TEXT LINE SSSSS.
11940 #                  SSSSS/X - TEXT LINE SSSSS HAS SYNTAX ERROR.
11950 #                  SSSSS/S - TEXT LINE SSSSS HAS A BAD SEQUENCE #.
11960 #                  TEXT IS NOT ALTERED IN ANY WAY.
11970 #
11980 #     2. RENUM SSSSS,IIIII,PPPPP,QQQQQ - CAUSE ALL TEXT LINES WHOSE
11990 #                  SEQ #'S ARE >= PPPPP AND <= QQQQQ TO BE ASSIGNED NEW SEQ #'S.
12000 #                  SSSSS IS THE FIRST NEW SEQ # ASSIGNED WITH SUBSEQUENT #'S
12010 #                  GENERATED BY ADDING IIIII TO THE SEQ # OF THE PREVIOUS TEXT LINE.
12020 #                  SSSSS AND IIIII MUST BE IN THE RANGE 1 - 65529 AND HAVE
12030 #                  DEFAULT VALUE 10. PPPPP MUST BE IN THE RANGE 1 - 65529,
12040 #                  HAS DEFAULT VALUE 0 AND MUST BE <= SSSSS.
12050 #                  QQQQQ MUST BE IN THE RANGE 1-65529, >= SSSSS, AND HAS DEFAULT
12060 #                  VALUE 65529.
12070 #
12080 # COMMANDS USING THE DEFAULTS ARE:
12090 #     RENUM ,

```

```

12100 # RENUM SSSSS
12110 # RENUM ,IIIII
12120 # RENUM ,PPPPP
12130 # RENUM ,QQQQQ
12140 # RENUM SSSSS,IIIII
12150 # RENUM SSSSS,PPPPP
12160 # RENUM ,IIIII,PPPPP
12170 # RENUM SSSSS,,QQQQQ
12180 # RENUM ,IIIII,QQQQQ
12190 # RENUM ,PPPPP,QQQQQ
12200 # RENUM SSSSS,IIIII,PPPPP
12210 # RENUM SSSSS,IIIII,QQQQQ
12220 # RENUM SSSSS,PPPPP,QQQQQ
12230 # RENUM ,IIIII,PPPPP,QQQQQ
12240 #
12250 # IF ANY ERROR ENCOUNTERED BEFORE TEXT IS ALTERED, THE COMMAND REVERTS
12260 # TO 'RENUM U' AND DISPLAYS ALL ERRORS IT CAN FIND.
12270 # IF AN ERROR ENCOUNTERED AFTER TEXT ALTERATION BEGINS, 'FATAL ERROR,
12280 # TEXT NOW BAD' IS DISPLAYED AND A 4030 EXIT TAKEN, GOING EITHER
12290 # TO 'DOS READY' OR TO DEBUG (IF CMD"D" ACTIVE").
12300 # THE BASIC TEXT MUST NOT BE RECLAIMED (DON'T USE BASIC *).
12310 #
12320 # IF 'SYS11/SYS' NOT IN THE SYSTEM WHEN COMMAND 'RENUM' IS EXECUTED,
12330 # BASIC ERROR 'UNPRINTABLE ERROR'(CODE 38 DECIMAL) WILL
12340 # BE INVOKED, WITH CMD"E" DISPLAYING 'FILE NOT IN DIRECTORY'.
12350 #
12360 #
12370 #
12380 #*** 047 *****
12390 #
12400 #BASIC IS ALTERED TO ALLOW L, E OR D IN PLACE OF LIST, EDIT OR DELETE
12410 #RESPECTIVELY WHEN EACH IS ALL OF THE FOLLOWING:
12420 # 1. 1ST CHAR OF INPUT LINE.
12430 # 2. FOLLOWED BY EITHER A PERIOD OR A DECIMAL DIGIT.
12440 # 3. THE INPUT LINE DOES NOT CONTAIN AN :.
12450 #
12460 #
12470 #
12480 #*** 048 *****
12490 #
12500 #PROGRAM DISKDUMP/BAS HAS BEEN UPGRADED TO ALLOW:
12510 # 1. OPTION OF SENDING DUMP TO PRINTER OR DISPLAY.
12520 # 2. OPTION OF DUMPING BY DISK RELATIVE SECTOR WITHIN FILE OR
12530 # LOGICAL RECORD # WITHIN FILE (RECORDS END WITH CARRIAGE
12540 # RETURN (ASC CODE 13 DECIMAL). EACH OCCURENCE OF THIS CHAR
12550 # IS TREATED AS AN EOR REGARDLESS OF OTHER SPECIAL MEANINGS
12560 # (IE, WITHIN A QUOTE STRING OR PRECEDED BY A DOWN-ARROW).
12570 # 3. A DUMP PAUSE COMMAND.
12580 #
12590 #
12600 #
12610 #*** 049 *****
12620 #
12630 #PROGRAM EDTASM/CMD HAS BEEN UPGRADED TO:
12640 # 1. READ SOURCE FROM DISK AS WELL AS CASSETTE.
12650 # 2. WRITE SOURCE AND/OR OBJECT TO DISK AS WELL AS CASSETTE. DISK
12660 # FILES ARE VALIDITY READ AFTER ALL SECTORS WRITTEN.
12670 # 3. ALLOW 'DOWN-ARROW' SCROLLING TO DISPLAY UP TO 15 TEXT LINES.
12680 # 4. PREVENT THE CONFUSING PRINTER OUTPUT ASSOCIATED WITH 'DEFM'.
12690 # ONLY THE 1ST BYTE OF ASSOCIATED OBJECT CODE IS LISTED.

```



```

12700 #      5. LIST SYMBOLS IN ALPHABETICAL ORDER WITH REFERENCE LIST.
12710 #      6. ACCEPTS AND CONVERTS LOWER CASE ALPHA TO UPPER.
12720 #
12730 # SUPPLEMENTAL INSTRUCTIONS FOR THE EDITOR-ASSEMBLER.
12740 #
12750 #      1. TO LOAD A SOURCE MODULE INTO THE TEXT BUFFER.
12760 #          1. L D:NNNNNNNN/TIT,PPPPPPPPID IF SOURCE FROM DISK.
12770 #          2. L T:NNNNNN IF SOURCE FROM CASSETTE.
12780 #      IF THE TEXT BUFFER ALREADY CONTAINS TEXT, THE QUERY "TEXT IN
12790 #      BUFFER. ARE YOU CONCATENATING???" APPEARS. REPLY 'N' IF NOT,
12800 #      CAUSING BUFFER TO BE SET EMPTY BEFORE NLW MODULL LOAD.
12810 #      REPLY 'Y' IF YOU ARE, CAUSING THE NEW SOURCE TO BE APPENDED ONTO
12820 #      THE END OF THE OLD. NO CONCERN IS SHOWN FOR OVERLAPPING SEQ #;
12830 #      THEREFORE YOU SHOULD EXECUTE AN 'N' EDTASH COMMAND UPON COMPLETION
12840 #      OF THE LOAD TO ASSURE A VALID SET OF ASCENDING SEQUENCE NUMBERS.
12850 #
12860 #      3. TO STORE A SOURCE MODULE:
12870 #          1. W D:NNNNNNNN/TIT,PPPPPPPPID IF SOURCE GOING TO DISK.
12880 #          2. W T:NNNNNN IF SOURCE GOING TO CASSETTE.
12890 #
12900 #      4. FOR A COMMANDS WITH 'NO' OPTION NOT SPECIFIED, RESPOND TO THE
12910 #      QUERY "OBJECT FILE TO DISK OR TAPE? REPLY D OR T"!
12920 #          1. T IF OBJECT GOING TO CASSETTE. NAME WILL COME FROM
12930 #              THE A COMMAND.
12940 #          2. D IF OBJECT GOING TO DISK. RESPOND TO THE QUERY
12950 #              "OBJECT FILESPEC?" WITH THE NNNNNNNN/TIT,PPPPPPPPID
12960 #              FILESPEC OF THE OBJECT MODULL. THE FILE WILL BE
12970 #              OPENED IMMEDIATELY, BUT NOT WRITTEN UNTIL END OF
12980 #              ASSEMBLY LISTING. THE NAME IN THE A COMMAND IS
12990 #              IGNORED.
13000 #      5. WHEN AN OUTPUT SOURCE OR OBJECT FILE IS OPENED, ONE OF THE
13010 #      FOLLOWING IS DISPLAYED:
13020 #          1. "FILE ALREADY EXISTS. USE IT???", REPLY 'Y' IF
13030 #              THIS IS YOUR INTENTION. OTHERWISE REPLY 'BREAK' TO
13040 #              TERMINATE THE W OR A COMMAND.
13050 #          2. "***** FILE NON-EXISTENT. REPLY "C" TO
13060 #              CREATE IT". REPLY 'C' IF THIS IS YOUR INTENTION.
13070 #              OTHERWISE REPLY 'BREAK' TO TERMINATE THE W OR A COMMAND.
13080 #
13090 #      6. DUE TO AN ERROR IN THE ORIGINAL DOS, EDTASH RUNS WITH
13100 #      INTERRUPTS DISABLED (EXCEPT WHEN RE-ENABLED BY DISK I/O) IN ORDER
13110 #      THAT USE OF 'BREAK' WILL FUNCTION PROPERLY.
13120 #      7. THIS EDTASH CAN EXECUTE IN A REGULAR TRS DOS ENVIRONMENT.
13130 #      8. THIS EDTASH REQUIRES A TRSDOS-LIKE OPERATING SYSTEM, AND USES
13140 #      LEVEL II KEYBOARD, DISPLAY AND PRINTER ROUTINES AND CONTROL BLOCKS.
13150 #      USERS ALTERING THE SYSTEM BEWARE!!!!
13160 #
13170 #
13180 # *** 050 *****
13190 #
13200 # BASIC1/CMD IS A MODIFIED VERSION OF THE ORIGINAL LEVEL I BASIC.
13210 # THIS MODULE LOADS FROM DISK INTO 4E00-7FFFH, AND CAN FUNCTION AS LEVEL I
13220 # FOR LEVEL I BASIC PROGRAMS THAT EXECUTE IN A REGULAR LEVEL I COMPUTER
13230 # WHERE "MEM SIZE" IS SPECIFIED AS 28160 (4000H) OR LESS. HOWEVER, DUE TO
13240 # THE WIDE VARIETY OF LEVEL I BASIC PROGRAMS, APPARAT DOES NOT GUARANTEE THE
13250 # OPERATION OF ANY OF THEM VIA BASIC1/CMD.
13260 #
13270 # THE FOLLOWING COMMANDS ARE ADDED TO LEVEL I:
13280 #      1. CSAVE* THIS CAUSES THE PROGRAM TEXT TO BE MOVED INTO MAIN
13290 #              MEMORY AT 8000H AND UP, RATHER THAN SENT TO THE

```

```

13300 # CASSETTE, 'RESET' AND THEN DOS COMMAND 'LVIDSKSL'
13310 # WITH THE 'S' OPTION WILL STORE THE LEVEL I PROGRAM
13320 # LOCATED IN THE 8000-UP REGION ONTO DISK.
13330 #
13340 # 2. CLOAD# FIRST, EXECUTE DOS COMMAND 'LVIDSKSL' WITH 'L' OPTION.
13350 # THE SPECIFIED LEVEL I PROGRAM WILL BE LOADED FROM
13360 # DISK INTO MAIN MEMORY BEGINNING AT 8000H. AT THE
13370 # CONCLUSION OF THIS LOAD, DOS COMMAND 'LEVEL1' WILL
13380 # AUTOMATICALLY BE EXECUTED, THUS BRING UP LEVEL I.
13390 # ISSUING LEVEL I COMMAND 'CLOAD#' WILL CAUSE THE
13400 # LEVEL I TEXT BUFFER TO BE LOADED FROM MAIN MEMORY
13410 # AT 8000H, RATHER THAN FROM CASSETTE.
13420 # 3. CSAVE### A JUMP IS TAKEN TO LOCATION 0 TO CAUSE A POWER-ON
13430 # LEVEL II START UP, BUT WITHOUT LOSING THE CONTENTS
13440 # OF MEMORY. THIS IS INTENDED AS AN AID TO THE
13450 # NON-DISK USER WHERE 'RESET' IN A SYSTEM WITHOUT
13460 # THE EXPANSION INTERFACE LEAVES THE USER STILL IN
13470 # LEVEL I. RESPONDING MEM SIZE = 28160 OR LESS
13480 # WILL PRESERVE LEVEL I IN MEMORY FOR SUBSEQUENT
13490 # STARTUP VIA:
13500 # 1. 'SYSTEM'
13510 # 2. '/28160'
13520 #
13530 #
13540 #
13550 #*** 051 ****
13560 #
13570 # 'LVIDSKSL' THIS OBJECT MODULE PROVIDES AN INTERFACE BETWEEN LEVEL-I-
13580 #EXECUTING-IN-LEVEL-II AND THE LEVEL I BASIC PROGRAMS STORED ON DISK.
13590 #
13600 # TO THE QUERY "REPLY "S" FOR SAVE, "L" FOR LOAD.", REPLY "S" IF A LEVEL I
13610 #BASIC PROGRAM IS IN MAIN MEMORY AT 8000H AS A RESULT OF A 'CSAVE#' LEVEL I
13620 # COMMAND SUCCESSFUL EXECUTION, AND IT IS TO BE STORED ON DISK. REPLY "L"
13630 #IF AN EXISTING LEVEL I PROGRAM ON DISK IS TO BE LOADED INTO MAIN MEMORY
13640 #AT 8000H, LEVEL I ACTIVATED VIA DOS COMMAND 'LEVEL1', AND THE PROGRAM TO
13650 # LOADED INTO LEVEL I'S TEXT BUFFER BY A SUBSEQUENT 'CLOAD#' ISSUED BY THE
13660 # USER.
13670 #
13680 #RESPOND TO THE QUERY "FILE-SPEC" WITH THE FILESPEC THE LEVEL I BASIC TEXT
13690 #MODULE HAS OR IS TO HAVE.
13700 #
13710 #IF "BAD TEXT PREFIX" DISPLAYED, ONE OF THE FOLLOWING EXISTS:
13720 # 1. TEXT FROM DISK IS NOT A LEVEL I BASIC TEXT MODULE.
13730 # 2. MAIN MEMORY IS NOT GOOD.
13740 # 3. USER FORGOT TO EXECUTE 'CSAVE#'.
13750 #
13760 # END.

```

```
00100 # 'DISASSEM'   VERSION 1.1
00110 #
00120 # THIS DISASSEMBLER DISASSEMBLES Z-80 CPU CODE, AND IS DESIGNED TO RUN ON
00130 # THE TRS-80 DOS SYSTEM USING AS SOURCE EITHER A STANDARD TRS-80 LOAD MODULE
00140 # ON DISK OR A PREVIOUSLY LOADED OBJECT MODULE IN MAIN MEMORY.
00150 # DISASSEMBLER WRITTEN IN Z-80 CODE
00160 #
00170 # OPERATION
00180 #
00190 # ENTER DOS COMMAND 'DISASSEM'
00200 #
00210 # RESPONSES TO QUERY 'OBJECT FROM MAIN MEMORY OR DISK'?
00220 #   1. NULL - SAME AS 'D'
00230 #   2. 'D' - OBJECT IS A DISK LOAD MODULE
00240 #       1. RESPONSE TO QUERY 'FILESPEC'?
00250 #           1. FILESPEC IN 'NNNNNNNN/TTT.PPPPPPPP:D' FORMAT
00260 #       2. RESPONSE TO QUERY 'OFFSET OBJECT VIRTUAL ADDRESSES BY (HEX)?'
00270 #           1. NULL - = 0
00280 #           2. 'NNNN' - 1 TO 4 HEX DIGIT VALUE WHICH WHEN ADDED TO LOAD
00290 #               ADDRESSES WITHIN THE LOAD MODULE WILL GIVE PROPER ADDRESS
00300 #               WHERE THE CURRENT INST BEING DISASSEMBLED WOULD BE DURING
00310 #               NORMAL EXECUTION OF THAT CODE.
00320 #               THIS PARAMETER IS NEEDED WHEN A OBJECT MODULE LOADS TO ONE
00330 #               PLACE IN MAIN MEMORY, BUT ACTUALLY EXECUTES FROM ANOTHER.
00340 #               WRAP-AROUND IS ALLOWED.
00350 #               EXAMPLE: IF THE OBJECT MODULE LOADS INTO C000-FFFF BUT IS
00360 #               TO EXECUTE IN 7000-AFFF, APPLYING AN OFFSET OF B000 WILL
00370 #               CAUSE THE DISASSEMBLER TO DISASSEMBLE AS IF THE LOAD
00380 #               WAS ACTUALLY DONE TO 7000-AFFF.
00390 #
00400 #       3. RESPONSE TO QUERY 'VIRTUAL RESTART LOCATION (HEX)?'
00410 #           1. NULL = NONE
00420 #           2. 'NNNN' - 1 TO 4 HEX DIGIT VIRTUAL RESTART LOCATION.
00430 #               ALLOWS RESTART OF A LARGE DISASSEMBLY WITHIN THE INSTRUCTION
00440 #               PRINT PORTION OF THE LISTING. 'NNNN' IS THE LISTED LOC OF ANY
00450 #               INSTRUCTION OF THE DISASSEMBLY, USUALLY CHOSEN AS THE LOCATION
00460 #               VALUE PRINTED FOR THE FIRST INST ON THE PAGE WHERE PRINTING
00470 #               WAS INTERRUPTED.
00480 #
00490 #   3. 'M' - OBJECT IS IN MAIN MEMORY
00500 #       1. RESPONSE TO QUERY 'OBJECT VIRTUAL BASE ADDRESS (HEX)?'
00510 #           1. 'NNNN' 1 TO 4 HEX DIGIT LOCATION VALUE WHERE THE
00520 #               OBJECT CODE IS CONSIDERED TO EXECUTE FROM. IN THE LISTING,
00530 #               THIS VALUE WILL BE THE FIRST INSTRUCTION PRINTED'S LOCATION
00540 #               VALUE.
00550 #       2. RESPONSE TO QUERY 'OBJECT REAL BASE ADDRESS (HEX)?'
00560 #           1. NULL REAL LOCATION SAME AS VIRTUAL
00570 #           2. 'NNNN' 1 TO 4 HEX DIGIT MEMORY LOCATION WHERE THE
00580 #               DISASSEMBLER WILL ACTUALLY FIND THE OBJECT CODE.
00590 #
00600 #
00610 # RESPONSES TO QUERY 'OUTPUT TO PRINTER'?
00620 #   1. 'Y' OUTPUT IS TO LINE PRINTER. ALL LINES < 64 CHARS.
00630 #       1. RESPONSE TO REQUEST 'REPLY 'ENTER' WHEN PRINTER AT TOP OF PAGE'?
00640 #           1. NULL SET PRINTER EXACTLY AT TOP OF PAGE. THIS PROGRAM ASSUMES
00650 #               66 LINES/PAGE AND LEAVES THE TOP 3 AND BOTTOM 3 LINES BLANK
00660 #               ON EVERY PAGE.
00670 #       2. IF OBJECT FROM MAIN MEMORY, RESPOND TO QUERY 'BYTE COUNT (HEX)?'
00680 #           1. 'NNNN' 1 TO 4 HEX DIGIT COUNT OF BYTES TO
00690 #               DISASSEMBLE
```

```

00700 #
00710 # 2. NULL - SAME AS 'N'
00720 # 3. 'N' - LISTINGS WILL GO TO DISPLAY. IF OBJECT FROM DISK, THE LOCATION
00730 # TABLE WILL NOT BE LISTED.
00740 # 1. RESPONSES TO QUERY 'NORMAL DISPLAY PAUSES'?
00750 # 1. NULL - SAME AS 'Y'
00760 # 2. 'Y' - DURING DISASSEMBLED INSTRUCTION LISTING, THE PROGRAM WILL
00770 # PAUSE AT THE END OF EACH DISPLAY PAGE, TO CONTINUE ON TO THE
00780 # NEXT PAGE, HIT 'ENTER'. A REPLY OF 'X' WILL TERMINATE THE
00790 # DISASSEMBLY.
00800 # 3. 'N' - THE DISASSEMBLY WILL DISPLAY AS FAST AS THE PROGRAM
00810 # CAN HOBBLE. TO PAUSE, HIT 'P'. TO CONTINUE AFTER PAUSE,
00820 # HIT 'ENTER'.
00830 #
00840 # RESPONSES TO QUERY 'ANY OPTIONS'? (OBJECT FROM DISK, ONLY)
00850 # 1. NULL - NO MORE OPTIONS
00860 # 2. 'NIP' - DO NOT PRINT OR DISPLAY THE DISASSEMBLED INSTRUCTIONS. THIS DOES
00870 # NOT AFFECT THE DISPLAY OF THE LOCATION-CROSS-REFERENCE PASS.
00880 # 3. 'RTD' - LOCATION REFERENCE TABLE IS TO BE STORED ON DISK.
00890 # AFTER THE LOCATION-CROSS-REFERENCE PASS, THE PROGRAM WILL QUERY
00900 # 'REFERENCE TABLE FILESPEC'? RESPOND WITH THE FILESPEC IN
00910 # NNNNNNNN/TTT.PPPPPPPP;D FORMAT IDENTIFYING THE FILE TO CONTAIN THE
00920 # REFERENCE TABLE. REFERENCE TABLE FILES ON DISK CAN BE USED (BY
00930 # USER-CREATED PROGRAM) TO MERGE THE REFERENCE TABLES OF TWO OR MORE
00940 # PROGRAMS. SEE BELOW FOR FILE FORMAT.
00950 # 4. 'REA' - ENABLE LISTING OF ALL TYPES OF REFERENCES; IS THE DEFAULT CONDITION.
00960 # 5. 'RE&' - ENABLE LISTING OF THE SPECIFIED REFERENCE TYPE WHERE & IS ONE OF
00970 # L, P, R, S, T, U, V, W OR X. REFERENCE TYPES ARE DEFINED AT THE
00980 # BEGINNING OF EACH LOCATION TABLE LISTING.
00990 # 6. 'RIA' - DISABLE LIST OF ALL TYPES OF REFERENCES
01000 # 7. 'RI&' - DISABLE LISTING OF THE SPECIFIED REFERENCE TYPE WHERE & IS ONE OF
01010 # L, P, R, S, T, U, V, W OR X.
01020 #
01030 #
01040 #
01050 # THE DISASSEMBLER OPERATES IN THESE PHASES:
01060 # 1. (OBJECT FROM DISK) BUILD LOCATION REFERENCE TABLE. IF INSUFFICIENT
01070 # MEMORY AVAILABLE, 'INSUFFICIENT MEMORY' WILL BE PRINTED AND THE
01080 # DISASSEMBLY TERMINATED.
01090 # DURING THIS PHASE, THE OBJECT MODULE WILL BE DISASSEMBLED TO OBTAIN
01100 # REFERENCE INFORMATION AND WILL DISPLAY THIS PASS WITHOUT PAUSE.
01110 # 2. WRITE REFERENCE TABLE FILE. (RTD OPTION ONLY)
01120 # 3. LIST DISASSEMBLED INSTRUCTIONS. IF SPEC'ED, DISPLAY PAUSES WILL OCCUR.
01130 # 4. PRINT LOCATION REFERENCE TABLE. (OBJECT FROM DISK ONLY AND PRINTER
01140 # USE SPEC'ED)
01150 #
01160 # IF THE DOS OPERATING SYSTEM RETURNS AN ERROR CODE, '&& DISK ERROR' MSG IS
01170 # DISPLAYED AND THE DISASSEMBLY TERMINATED. && IS A TWO HEX DIGIT ERROR CODE.
01180 # SOME OF THESE CODES ARE:
01190 # 1C OR 1D - EOF (OBJECT NOT A LOAD MODULE ?????)
01200 # 25 OBJECT IS READ-PROTECTED
01210 # 1B NO MORE SPACE ON DISKETTE
01220 # 19 OBJECT IS PASSWORD PROTECTED
01230 # 18 OBJECT FILE DOES NOT EXIST
01240 #
01250 # 'DISK OBJECT FILE FORMAT NOT AS EXPECTED' MSG WILL BE DISPLAYED IF THE
01260 # DISASSEMBLER FINDS SOMETHING WRONG WITH THE OBJECT MODULE.
01270 #
01280 # WHILE INSTRUCTION DISPLAYING OR PRINTING IS IN PROGRESS, HOLDING DOWN THE 'X'
01290 # OR THE 'P' KEY WILL CAUSE DISASSEMBLY TERMINATION OR PAUSE RESPECTIVELY.

```

```

01300 ; HIT THE 'ENTER' KEY TO CONTINUE AFTER PAUSE.
01310 ;
01320 ;SUFFIXED TO EACH REFERENCING LOCATION VALUE IS A REFERENCE TYPE CODE (DEFINED
01330 ; AT TOP OF EACH REFERENCE LISTING) GIVING THE TYPE OF Z-80 INSTRUCTION MAKING
01340 ;THE REFERENCE.
01350 ;
01360 ;ON THE DISASSEMBLED INSTRUCTION PRINT LINES:
01370 ;     1. COLUMN 1 INDICATES THE NUMBER OF REFERENCES TO BYTES OF THE INSTRUCTION.
01380 ;     VALUE IS HEXIDEcimal WITH BLANK : 0 AND F MEANS 15 OR MORE REFERENCES.
01390 ;     2. COLUMN 2 INDICATES WHICH BYTES OF THE INSTRUCTION HAVE BEEN REFERENCED.
01400 ;     IF BLANK AND COLUMN 1 NON-BLANK, THEN ONLY THE INST'S 1ST BYTE REFERENCED.
01410 ;     OTHERWISE THE HEX DIGIT REPRESENTS A 4-BIT BINARY MASK OF WHICH
01420 ;     BYTES, FROM THE LEFT, ARE REFERENCED.
01430 ;
01440 ;
01450 ; PROGRAM WARRANTY IS LIMITED TO THE PROVISION ON EITHER DISK OR TAPE (SELLER'S
01460 ;DISCRETION) AN OPERATIONAL LOAD MODULE TO RUN UNDER TRS-80 DOS 1.1
01470 ;THERE IS NO WARRANTY FOR CONTINUED OPERATION UNDER SUBSEQUENT DOS RELEASES.
01480 ;
01490 ; THIS PROGRAM LOADS INTO 8200, THEN SHIFTS ITSELF TO 5200 FOR EXECUTION.
01500 ;ENTRY POINT = 9B00H.
01510 ;
01520 ;TO LOAD THIS MODULE FROM TAPE TO DISK, USE TRS-80'S 'TAPEDISK' PROGRAM.
01530 ;NAME ON TAPE IS 'DISASM'
01540 ;NAME ON DISK IS 'DISASSEM/CMD'
01550 ;
01560 ;
01570 ;FORMAT OF REFERENCE TABLE FILE CREATED BY 'RTD' OPTION:
01580 ;     1. 1 BYTE = 00H, BACKWARD EOF, IGNORE IT.
01590 ;     2. 1 OR MORE ENTRIES OF THE FORM:
01600 ;         1. 'LOCATION' LOW VALUE BYTE
01610 ;         2. 'LOCATION' HIGH VALUE BYTE
01620 ;         3. CONTROL BYTE, BITS 7-0 (7 IS LEFTMOST)
01630 ;             7-6 = 11: DUMMY LAST ENTRY IN TABLE, IGNORE ALL OTHER
01640 ;             BITS AND BYTES OF ENTRY.
01650 ;             7-6 = 01: REFERENCEE ENTRY:
01660 ;                 1. BITS 5-0 = 0.
01670 ;                 2. 'LOCATION' REFERENCED BY THE ONE OR MORE
01680 ;                 FOLLOWING REFERENCOR ENTRIES.
01690 ;             7-6 = 00: REFERENCOR ENTRY:
01700 ;                 1. THE INSTRUCTION AT THIS 'LOCATION' REFERENCED
01710 ;                 'LOCATION' IN THE PREVIOUS REFERENCEE ENTRY
01720 ;                 2. BITS 5-0 = 00 IFH CODE INDICATING TYPE OF
01730 ;                 INSTRUCTION MAKING THE REFERENCE:
01740 ;                     0 = S
01750 ;                     1 = I
01760 ;                     2 = U
01770 ;                     3 = V
01780 ;                     4 = W
01790 ;                     5 = X
01800 ;                     6-7 UNUSED
01810 ;                     8 = P
01820 ;                     9 = L
01830 ;                     A = R
01840 ;                     B-1F UNUSED
01850 ; SEE A REFERENCE TABLE LISTING FOR DEFINITIONS.
01860 ;
01860 ;     END.

```

```
00100 # PROGRAM 'LMOFFSET' VERSION 1.1
00110 #
00120 # PROGRAM RESIDES IN 5200-5FFF.
00130 #
00140 # PROGRAM FUNCTIONS AS FOLLOWS:
00150 # 1. READS EITHER A TAPE-TYPE ASSEMBLY LOAD MODULE FROM TAPE OR A DISK-TYPE
00160 # ASSEMBLY LOAD MODULE FROM DISK
00170 # 1. IF SOURCE FROM DISK, PROGRAM ASKS FOR SOURCE FILESPEC IN
00180 # DDDDDDDDD/TTT.PPPPPPPP:D FORMAT
00190 # 2. DISPLAYS:
00200 # 1. AREA INTO WHICH MODULE WILL LOAD WHEN WRITTEN TO DISK.
00210 # 2. POSSIBLE CONFLICTS WITH SYSTEM STORAGE.
00220 # 3. MODULE ENTRY POINT. IF APPENDAGE HAS BEEN APPLIED, THE
00230 # ENTRY POINT WILL BE INTO THE 15 BYTE APPENDAGE.
00240 # 3. ASKS FOR NEW LOAD POINT. REPLY EITHER WITH NEW LOAD POINT OR SIMPLY
00250 # REPLY 'ENTER' IF SATISFIED WITH CURRENT LOAD POINT.
00260 # 4. IF NEW LOAD POINT SPEC'ED, DIFFERENT FROM THE ORIGINAL LOAD POINT,
00270 # THE PROGRAM ASKS IF THE
00280 # APPENDAGE IS TO BE SUPPRESSED. IF SO, THE RESULTING MODULE CAN
00290 # ONLY BE USED VIA DOS 'LOAD' COMMAND AS THERE IS NO APPENDAGE TO
00300 # MOVE THE PROGRAM TO ITS EXECUTION AREA. RESULTING OUTPUT LOAD
00310 # MODULE CAN BE USED VIA 'LOAD' WHERE TWO OR MORE LOAD MODULES ARE
00320 # LOADED INTO MAIN MEMORY AND THEN STORED AS ONE LOAD MODULE VIA 'DUMP'.
00330 # 5. IF NEW LOAD POINT SPEC'ED, GO BACK TO #2 ABOVE.
00340 # 6. IF AT LEAST ONE NEW LOAD POINT SPEC'ED AND APPENDAGE NOT SUPPRESSED,
00350 # PROGRAM ASKS IF INTERRUPTS
00360 # ARE TO BE DISABLED. IF REPLY = 'Y' PROGRAM INSERTS A 'DI' (DISABLE
00370 # INTERRUPTS INSTRUCTION) INTO THE APPENDAGE AS ITS ONLY ATTEMPT TO
00380 # PREVENT INTERRUPTS FROM OCCURRING.
00390 # 7. ASKS FOR FILESPEC OF DISK-TYPE LOAD MODULE TO BE CREATED. REPLY WITH
00400 # FILESPEC IN NNNNNNNN/TTT.PPPPPPPP:D FORMAT.
00410 # 8. WRITES THE NEW DISK-TYPE LOAD MODULE TO DISK. IF A NEW LOAD POINT
00420 # WAS SPEC'ED:
00430 # 1. THE LOAD ADDRESS FOR EACH OBJECT CODE RECORD IS ALTERED.
00440 # 2. IF APPENDAGE NOT SUPPRESSED, AN EXTRA OBJECT CODE RECORD
00450 # (CALLED THE APPENDAGE) IS INSERTED
00460 # BEFORE THE ENTRY POINT RECORD.
00470 # 3. THE ENTRY POINT FOR THE MODULE BECOMES:
00480 # 1. IF APPENDAGE SUPPRESSED, ENTRY POINT = 0000H.
00490 # 2. OTHERWISE ENTRY POINT IS TO FIRST BYTE OF APPENDAGE.
00500 #
00510 # IF THE MODULE LOAD POINT HAS BEEN CHANGED AND APPENDAGE NOT SUPPRESSED,
00520 # THE MODULE IS EXTENDED BY A 15
00530 # BYTE APPENDAGE WHICH IS EXECUTED WHEN CONTROL IS PASSED TO THE MODULE ENTRY
00540 # POINT. THE APPENDAGE EXECUTION IS AS FOLLOWS:
00550 # 1. EXECUTES A DISABLE INTERRUPTS INSTRUCTION, IF SPEC'ED WHEN APPENDAGE
00560 # CREATED.
00570 # 2. MOVES THE OBJECT CODE (LESS THE APPENDAGE) FROM THE 'NEW' LOAD
00580 # AREA TO THE 'OLD' LOAD AREA.
00590 # 3. PASSES EXECUTION CONTROL TO THE 'OLD' ENTRY POINT
00600 #
00610 # THIS PROGRAM DOES NOT PERFORM ANY OBJECT CODE RELOCATION!!!!
00620 # IT ONLY MOVES CODE TO NEW LOAD LOCATIONS (APPLYING THE APPENDAGE) SO THAT DOS
00630 # CAN LOAD THE MODULE FROM DISK AND PASS CONTROL TO IT. IF THE PROGRAM EXECUTES
00640 # FROM DOS CONFLICTING AREA, IT MUST BE CAREFUL NOT TO RE-ENABLE THE INTERRUPTS.
00650 #
00660 # IF THE SOURCE PROGRAM LOADS INTO THE DISPLAY AREA (3C00-3FFF) WITHOUT OVERFLOWING
00670 # 17, THOSE OBJECT CODE RECORDS WILL NOT HAVE THEIR LOAD ADDRESSES MODIFIED.
00680 #
00690 # 4000-51FF MEMORY IS DOS. ANY PART OF THE TARGET PROGRAM EXECUTING FROM THERE
```

00700 ; WILL CAUSE DOS AND/OR TARGET PROGRAM MALFUNCTION.
00710 ;
00720 ;5200-6FFF MEMORY IS WHERE DOS COMMANDS EXECUTE FROM. YOUR DOS FUNCTIONS- USING
00730 ;PROGRAM MAY HAVE ITS LOAD POINT IN THIS REGION AND DOS WILL LOAD AND EXECUTE
00740 ;THE PROGRAM AS A COMMAND, BUT THE DOS COMMAND 'LOAD' WILL REJECT THE MODULE.
00750 ;
00760 ;7000-FFFF. NO PROBLEMS, OTHER THAN LACK OF RAM, FOR PROGRAM LOADING IN THIS AREA.
00770 ;
00780 ;WARNING!!!!!!
00790 ;FOR EACH TIME A GIVEN LOAD MODULE IS USED AS THE INPUT SOURCE AND A NEW LOAD
00800 ; POINT IS SPECIFIED, AN ADDITIONAL APPENDAGE IS APPENDED.
00810 ; THIS PROGRAM KNOWS AND CARES NOTHING ABOUT PREVIOUSLY EXISTING APPENDAGES.
00820 ;
00830 ;DURING SOURCE READ, THIS PROGRAM LOADS THE OBJECT MODULE INTO A CONTROL TABLE
00840 ;STARTING AT LOCATION 6000H. (VIEWABLE VIA 'DEBUG').
00850 ;IF THE OBJECT MODULE IS FROM DISK, THE DATA STORED IN THE TABLE
00860 ; IS EXACTLY THAT FOUND ON DISK. IF THE OBJECT MODULE IS FROM TAPE, IT IS CONVERTED
00870 ; TO DISK OBJECT MODULE FORMAT WHEN STORED IN THE TABLE. THE TABLE HAS THE
00880 ; FOLLOWING LOGICAL RECORD FORMATS:
00890 ; 1. OBJECT CODE RECORD.
00900 ; 1. ID = 01
00910 ; 2. 1 BYTE COUNT OF OBJECT BYTES (2. (0 - 2 : 256 - 258)
00920 ; 3. 2 BYTE LOAD ADDRESS
00930 ; 4. OBJECT BYTES
00940 ; 2. ENTRY POINT RECORD.
00950 ; 1. ID = 02
00960 ; 2. 02H BYTE COUNT
00970 ; 3. 2 BYTE ENTRY ADDRESS
00980 ; 3. COMMENT RECORD.
00990 ; 1. 1 BYTE ID.
01000 ; 1. ID = 0 OR 3 - FF IF MODULE FROM DISK
01010 ; 2. ID = 0 OR 3 - FD IF MODULE FROM TAPE
01020 ; 2. 1 BYTE COMMENT BYTE COUNT. 0 - 256.
01030 ; 3. COMMENT BYTES
01040 ; 4. (TAPE INPUT ONLY) EXTRANEIOUS PREFIX BYTES REC. THIS REC CONTAINS ALL
01050 ; DATA BYTES ENCOUNTERED ON THE TAPE BEFORE THE 'U' (55H) ID BYTE
01060 ; ENCOUNTERED.
01070 ; 1. ID = FF.
01080 ; 2. 2 BYTE LENGTH COUNT
01090 ; 3. EXTRANEIOUS BYTES
01100 ; 5. (TAPE INPUT ONLY) NAME RECORD. CONTAINS ALL BYTES FROM THE 'U' SYNCH
01110 ; BYTE TO EITHER A 3CH BYTE (ID FOR TAPE OBJECT CODE RECS) OR A 78H
01120 ; BYTE (ID FOR TAPE ENTRY POINT REC) ENCOUNTERED.
01130 ; 1. ID = FE.
01140 ; 2. 2 BYTE LENGTH COUNT
01150 ; 3. NAME BYTES
01160 ; 6. (TAPE INPUT ONLY) IMBEDDED EXTRANEIOUS DATA REC. CONTAINS EXTRANEIOUS DATA
01170 ; APPEARING BETWEEN END OF ONE OBJECT CODE REC AND EITHER THE NEXT
01180 ; OBJECT CODE RECORD OR THE ENTRY POINT RECORD.
01190 ; 1. ID = FD.
01200 ; 2. 2 BYTE LENGTH COUNT.
01210 ; 3. EXTRANEIOUS DATA BYTES
01220 ; 7. (TAPE INPUT ONLY) BAD CHECKSUM RECORD. FOLLOWS AN OBJECT CODE RECORD
01230 ; WHOSE CHECKSUM COMPUTED BAD.
01240 ; 1. ID = FC.
01250 ; 2. 1 BYTE CHECKSUM FROM TAPE.
01260 ; 3. 1 BYTE COMPUTED CHECKSUM DURING OBJECT RECORD READ.
01270 ;
01280 ; DURING READ FROM TAPE:
01290 ; 1. SINGLE '*' DISPLAYED WHEN READY FOR TAPE. DO REWIND (IF NECESSARY), FAST


```

01300 ;      FORWARD POSITIONING (IF NECESSARY) AND 'PLAY'.
01310 ;      2. '***' APPEARS WHEN TAPE READ SYNCHRONIZATION HAS COMPLETED.
01320 ;      3. ALTERNATES '*' AND BLANK IN RIGHT-MOST DISPLAY POSITION AS EACH OBJECT
01330 ;      RECORD READ COMPLETES WITH GOOD CHECKSUM.
01340 ;      4. DISPLAYS 'C' WHEN A BAD CHECKSUM ENCOUNTERED.
01350 ;      5. DISPLAYS 'P' IF LEADING EXTRANEIOUS DATA BYTES ENCOUNTERED.
01360 ;      6. DISPLAYS 'I' IF IMBEDDED EXTRANEIOUS DATA BYTES ENCOUNTERED.
01370 ;
01380 ;      END.
00100 ;
00110 ;
00120 ;
00130 ;
00140 ;
00150 ;
00160 ;

```

DOCUMENTATION FOR APPARAT 'SUPERZAP' 2.0

***** WARNING - PROGRAM MALFUNCTIONS IF ANY WRITE OPERATIONS
ATTEMPTED TO A WRITE-PROTECTED DISKETTE. *****

FUNCTION 'DD' DISPLAYS A DISK SECTOR. FUNCTION 'DM' DISPLAYS A
MODULO 256 BYTE BLOCK OF MAIN MEMORY RESPECTIVELY.
FOR 'DD', THE ERROR CODE RETURNED BY THE SYSTEM IS DISPLAYED
IN COLUMN 7 OF THE LAST LINE. IF = 6, THE READ WAS SUCCESSFUL
BUT THE SECTOR IS READ-PROTECTED (IE, NOT READABLE BY NORMAL
USER DISK OPERATIONS, BUT WAS READ BY SUPERZAP).

FOR 'DD' AND 'PD' FUNCTIONS, COLUMNS 1-6 ARE DEFINED:

- 1 - RELATIVE DISK NUMBER (0-3).
- 2-3 - RELATIVE TRACK NUMBER (HEX 0 - 22).
- 4 - RELATIVE SECTOR NUMBER ON TRACK (0 - 9).
- 5-6 - RELATIVE POSITION (HEX) WITHIN THE SECTOR.

WHEN A 'DD' SECTOR OR A 'DM' BLOCK IS DISPLAYED ON THE SCREEN,
THE PROGRAM IS CONSTANTLY MONITORING THE INPUT KEYS LOOKING
FOR ONE OF THE FOLLOWING COMMANDS:

- 'X' - TERMINATE THE PRIMARY FUNCTION.
- 'R' - DISPLAY AGAIN THE SAME SECTOR/BLOCK.
- '+' OR 'F' - SCROLL FORWARD ONE SECTOR/BLOCK. (LINE 1400 V FOR I=164387016442;
FOR CONTINUOUS ON HOLD DOWN
↓ POKES 0,1 NEXT)
- '-' OR '=' - SCROLL BACKWARD ONE SECTOR/BLOCK.
- 'J' - RESTART THE SAME PRIMARY FUNCTION.
- 'K' - 'DD' ONLY. SAME AS 'J' EXCEPT NEXT SECTOR FROM
(LINE 13 VER 3.0)
SAME DRIVE AS PREVIOUS DISPLAYED SECTOR.
- 'MOD&&' - MODIFY THE CURRENT SECTOR/BLOCK. && IS THE
RELATIVE POSITION WITHIN THE SECTOR/BLOCK OF
BYTE TO BE MODIFIED (HEX 00 - FF).

THE PROGRAM RESPONDS BY PLACING AN 'M' IN
COLUMN 7 OF THE MODIFICATION LINE. FURTHER
IT PRECEDES THE HEX DIGIT QUAD CONTAINING
THE NEXT HEX DIGIT TO BE CHANGED WITH AN
INDICATOR OF WHICH DIGIT IT IS, USING
'+' , '-' , '*' AND '/' FOR THE 1ST,
2ND, 3RD AND 4TH DIGITS OF THE QUAD
RESPECTIVELY.

RESPONSES WHILE IN 'MODIFICATION MODE' ARE:

- HEX DIGIT 0 - F - REPLACES CURRENT DIGIT AND
ADVANCES TO NEXT DIGIT.

'DM' REPLACEMENTS ARE FOR REAL. 'DD' REPLACEMENTS ARE BUFFERED UNTIL EITHER AN ENTER OR A 'Q' COMMAND. WHEN THE LAST DIGIT DISPLAYED IS REPLACED, NEXT POSITION INCREMENT WRAPS AROUND TO THE FIRST DIGIT, BUT NO FURTHER DIGIT REPLACEMENTS ARE ACCEPTED UNTIL AFTER A NON-HEX-DIGIT KEY IS PRESSED.

SPACE - CURRENT DIGIT IS NOT CHANGED AND POSITION IS ADVANCED 1 DIGIT.
RIGHT-ARROW - SAME AS 'SPACE'.

SHIFT-LEFT-ARROW - CURRENT DIGIT NOT CHANGED AND POSITION IS RETARDED 4 DIGITS.
LEFT-ARROW - CURRENT DIGIT NOT CHANGED AND POSITION IS RETARDED 1 DIGIT.
SHIFT-RIGHT-ARROW - CURRENT DIGIT NOT CHANGED AND POSITION IS ADVANCED 4 DIGITS.
UP-ARROW - CURRENT DIGIT NOT CHANGED AND POSITION IS RETARDED 1 LINE, FIRST LINE WRAPS TO LAST.
DOWN-ARROW - CURRENT DIGIT NOT CHANGED AND POSITION ADVANCED 1 LINE, LAST LINE WRAPS TO FIRST.

'Q' - ('DD' ONLY). CANCELS WRITE OF UPDATED SECTOR. TERMINATES 'MODIFICATION MODE'.
ENTER - TERMINATES 'MODIFICATION MODE'. FOR 'DD', THE UPDATED SECTOR IS WRITTEN BACK TO DISK.

'SCOPY' - ('DD' ONLY) MOVE THE DISPLAYED SECTOR TO A DISK LOCATION TO BE SPECIFIED.

THE USER WILL BE ASKED TO RESPOND 'Y' OR 'N' AS TO THE TO-BE READ PROTECT STATE OF THE DESTINATION SECTOR.

FUNCTIONS 'PD' AND 'PM' SEND THE DISPLAYED BLOCKS TO THE PRINTER.

TO HALT PRINTER ACTION, HOLD DOWN THE 'H' KEY UNTIL THE PRINTER STOPS. THIS ACTION TERMINATES THE FUNCTION.

FOR PRINTER OPERATIONS YOU MUST SPECIFY A BYTE COUNT (FOR 'PM') OR A SECTOR COUNT (FOR 'PD') SO PROGRAM WILL KNOW WHEN TO STOP PRINTING.

THE 'ZERO DISK SECTORS' FUNCTION DOES JUST THAT!!!!!!

IF A SECTOR BEING ZEROED IS READ-PROTECTED, THE USER IS ASKED (REPLY 'Y' OR 'N') IF HE WANTS THAT SECTOR TO REMAIN READ-PROTECTED.

'COPY DISK SECTORS' FUNCTION COPIES THE SPECIFIED SECTORS TO NEW DISK LOCATIONS, DUPLICATING INTO THE NEW LOCATIONS THE READ-PROTECT STATUS OF THE OLD.

NORMALLY, THE COPY PROCEEDS IN ASCENDING TRACK/SECTOR # ORDER.

```

06300 ; THE RIGHT, LEFT, UP, AND DOWN ARROWS MAY BE USED TO
06400 ; ALTER POSITION WITHIN SECTOR WITHOUT CHANGING CONTENTS.
06500 ; 13. TO ALTER THE CURRENT HEX DIGIT, SIMPLY DEPRESS THE KEY
06600 ; FOR THE NEW VALUE, 0-9, A-F. THE UPDATE POSITION WILL BE
06700 ; AUTOMATICALLY ADVANCED 1 HEX DIGIT POSITION
06800 ; 14. TYPE IN ALL THE NEW HEX DIGIT VALUES.
06900 ; 15. IF NOT SATISFIED WITH THE CHANGES TO THE SECTOR, PRESS Q
07000 ; TO CANCEL THE MODIFICATION AND TO REDISPLAY THE SECTOR.
07100 ; 16. WHEN SATISFIED WITH THE CHANGES TO THE SECTOR, PRESS ENTER,
07200 ; AND WHEN INSTRUCTED, PRESS ENTER AGAIN.
07300 ; THE NEW SECTOR CONTENTS WILL BE DISPLAYED.
07400 ; 17. WHEN A SECTOR IS DISPLAYED AND YOU ARE NOT IN MODEB STATE
07500 ; 1. PRESS J IF YOU WISH TO DISPLAY ANOTHER SECTOR (TO
07600 ; STEP 6 ABOVE)
07700 ; 2. TO MODIFY TO CURRENT SECTOR, GO TO STEP 12 ABOVE.
07800 ; 18. TO EXIT SUPERZAP, PRESS BREAK
07900 ; 19. RUN BASIC PROGRAM 'SUPERZAP.COM' IF YOU WISH TO KNOW MORE
08000 ; OF SUPERZAP'S OPERATION
08100 ;
08200 ;
08300 ;
08400 ;
08500 ; ALL MANDATORY ZAPS ARE DATED AS TO THE DATE APPARAT INCORPORATED THE
08600 ; ZAP INTO MODULES BEING SUBSEQUENTLY SHIPPED.
08700 ;
08800 ;
08900 ; ALL OPTIONAL ZAPS ARE DATED AS TO WHEN THE CHANGE BECAME AVAILABLE
09000 ;
09100 ; SOME EARLY PRE-RELEASE COPIES OF NEW DOS DO NOT HAVE THESE ZAPS IN
09200 ; THE EXACT LOCATIONS.
09300 ;
09400 ; BEFORE MAKING ANY PART OF A ZAP, BACKUP YOUR NEWDOS, AND THEN VERIFY THAT
09500 ; ALL PARTS OF THE ZAP ARE MAKEABLE.
09600 ;
09700 ; ***** ZAP 001 ***** 03/16/79 *****
09800 ;
09900 ; MANDATORY ZAP TO INCREASE TRACK FORMATING TOLERANCE IN FULL DISKETTE 'COPY'.
10000 ;
10100 ; DISK 012000 = START OF SYS3/SYS FILE
10200 ; CHANGE DISK 0123F7 CONTENTS FROM 'FE4938' TO 'FE2738' ✓
10300 ;
10400 ;
10500 ;
10600 ; ***** ZAP 002 ***** 03/16/79 *****
10700 ;
10800 ; OPTIONAL ZAP TO DISABLE THE EFFECTS OF CHANGE #018 (SEE NEWDOS DOCUMENTATION)
10900 ; TO ALLOW LOWER CASE LETTERS (KEYED IN AS UPPER) TO APPEAR IN BASIC DIRECT OR
11000 ; TEXT STATEMENTS PLEASE NOTE THAT CHANGE #018 OF NEWDOS NEVER DID AFFECT
11100 ; DATA INPUT VIA 'INPUT' OR 'INKEY#'.
11200 ;
11300 ; DISK 003500 = START OF FILE 'BASIC/END'
11400 ; CHANGE DISK 00356C CONTENTS FROM '020052C3' TO '033360C3' ✓
11500 ;
11600 ;
11700 ;
11800 ; ***** ZAP 003 ***** 03/23/79 *****
11900 ;
12000 ; MANDATORY ZAP TO MAKE NEWDOS BASIC FUNCTION 'REF' WORK PROPERLY WITH 'FIELD'.
12100 ;
12200 ; DISK 022500 = START OF FILE 'SYS12.SYS'
12300 ; CHANGE DISK 022779 CONTENTS FROM '301577' TO '102H77' ✓
12400 ; CHANGE DISK 022798 CONTENTS FROM '3004FE' TO '2000FE' ✓
12500 ; CHANGE DISK 022799 CONTENTS FROM '101101' TO '101000' ✓

```

```

12600 ;CHANGE DISK 0227A0 CONTENTS FROM '2807EB' TO '28F8EB' ✓
12700 ;CHANGE DISK 0227A3 CONTENTS FROM '1804FE' TO '18EDFE' ✓
12800 ;CHANGE DISK 0227B3 CONTENTS FROM '0A154F' TO '0A6E4F' ✓
12900 ;CHANGE DISK 0227D1 CONTENTS FROM '18BF79' TO '18B379' ✓
13000 ;
13100 ;
13200 ;
13300 ;***** ZAP 004 ***** 04/03/79 *****
13400 ;
13500 ;MANDATORY ZAP TO CORRECT LOADER 'FILE ACCESS DENIED' ERRORS FOR
13600 ;EXECUTE ONLY ATTRIBUTED MODULES.
13700 ;
13800 ;DISK 000500 = START OF FILE SYS0/SYS
13900 ;CHANGE DISK 0014EB CONTENTS FROM
14000 ; C9 13 1A 1B E6 07 FE 07 38 07 3E 25 B7 C9 00 00 00 ED
14100 ; TO
14200 ; C9 62 6B 23 7E E6 07 FE 07 38 04 3E 25 B7 C9 0B 96 ED ✓
14300 ;
14400 ;
14500 ;
14600 ;***** ZAP 005 ***** 04/03/79 *****
14700 ;
14800 ;MANDATORY ZAP TO ALLOW 'COPY' TO WORK WITH PASSWORDS ENABLED.
14900 ;
15000 ;THIS CHANGE CHANGES THE 17-20TH BYTES OF FILE 'COPY/CMD'S' FPOE (THE UPDATE
15100 ;AND ACCESS PASSWORD ENCODES. THIS FPOE EXISTS SOMEWHERE IN TRACK 11 HEX.
15200 ;RELATIVE SECTORS 2-9. THE FOLLOWING DISK ADDRESS IS ONLY WHERE THE CHANGE
15300 ;TAKES PLACE FOR A PARTICULAR SYSTEM. YOUR LOCATION ON THE DISKETTE MAY
15400 ;BE DIFFERENT BUT THE CHANGE IS THE SAME.
15500 ;
15600 ;CHANGE DISK 0113D0 CONTENTS FROM
15700 ; EB 29 21 0E
15800 ; TO
15900 ; 81 30 96 42 ✓
16000 ;
16100 ;
16200 ;
16300 ;***** ZAP 006 ***** 04/03/79 *****
16400 ;
16500 ;OPTIONAL ZAP TO PERMANENTLY DISABLE 'DEBOUNCE', ELIMINATING NEED TO
16600 ;HOLD SHIFT-UP-ARROW DURING RESET/POWER ON.
16700 ;
16800 ;DISK 000500 = START OF FILE 'SYS0/SYS'
16900 ;CHANGE DISK 0015F1 CONTENTS FROM
17000 ; 09 28 06
17100 ; TO
17200 ; 09 18 06
17300 ;
17400 ;
17500 ;
17600 ;***** ZAP 007 ***** 04/03/79 *****
17700 ;
17800 ;MANDATORY ZAP TO CORRECT CMD"%" MALFUNCTION WHEN INSUFFICIENT FREE MEMORY
17900 ;AVAILABLE TO 'SAVE BASIC'S DATA.
18000 ;
18100 ;DISK 003500 = START OF FILE 'BASIC/CMD'
18200 ;CHANGE DISK 00387A CONTENTS FROM '0A 7A 19 D1' TO '0A 1D 55 D1' ✓
18300 ;CHANGE DISK 003887 CONTENTS FROM '02 7A 19 E1' TO '02 1D 55 E1' ✓
18400 ;CHANGE DISK 003941 CONTENTS FROM
18500 ; 57 5A C3 9A 5B C3 99 5B C3 65 5B C3 84 57
18600 ; TO
18700 ; 57 21 7A 19 E5 01 00 01 C5 C5 C7 F7 54 57 ✓
18800 ;

```

N o

Supplemental Instructions for
Editor-Assembler 1.1
For Disk Operations

1. Load the editor-assembler from disk via the DOS command: EDTASM
2. To load a source module into the text buffer.
 1. L D=nnnnnnnn/ttt.pppppppp:d if source from disk
 2. L T=nnnnnn if source from tape

When an L command is issued and previous text exists in the buffer, normal editor-assembler operations assume the user wants to concatenate the new source onto the end of the old. However, under the modifications, ~~if the source is from disk,~~ the query "TEXT IN BUFFER. ARE YOU CONCATENATING????"/^{appears} Reply N if text buffer is to be cleared before the load or Y if concatenation is to occur. If concatenation is done, either from tape or disk, an N command should be issued after the L command conclusion to assure a valid set of ascending sequence numbers.

3. To store a source module:
 1. W D=nnnnnnnn/ttt.pppppppp:d if source is to go to disk
 2. W T=nnnnnn if source is to go to tape
4. For A commands with NO option not specified:
 1. Respond to the query "OBJECT FILE TO DISK OR TAPE? REPLY D OR T"
 1. T if object to go to tape. Name will come from the A command.
 2. D if object to go to disk. Respond to query "OBJECT FILESPEC?" with the nnnnnnnn/ttt.pppppppp:d filespec of the object module. The file will be opened immediately, but not written until end of assembly listing. The name in the A command is ignored.