```
CP/M MACRO ASSEM 2.0    #001    SYSGEN - SYSTEM GENERATION PROGRAM 8/79

                                TITLE   'SYSGEN - SYSTEM GENERATION PROGRAM 8/79'
                        ;       SYSTEM GENERATION PROGRAM, VERSION FOR MDS
0014 =                  VERS    EQU     20      ;X.X
                        ;
                        ;       COPYRIGHT (C) DIGITAL RESEARCH
                        ;                1976, 1977, 1978, 1979
                        ;
001A =                  NSECTS  EQU     26      ;NO. OF SECTORS PER TRACK
0002 =                  NTRKS   EQU     2       ;NO. OF OPERATING SYSTEM TRACKS
0004 =                  NDISKS  EQU     4       ;NUMBER OF DISK DRIVES
0080 =                  SECSIZ  EQU     128     ;SIZE OF EACH SECTOR
0007 =                  LOG2SEC EQU     7       ;LOG 2 SECSIZ
0001 =                  SKEW    EQU     1       ;SECTOR SKEW FACTOR
                        ;
005C =                  FCB     EQU     005CH   ;DEFAULT FCB LOCATION
007C =                  FCBCR   EQU     FCB+32  ;CURRENT RECORD LOCATION
0100 =                  TPA     EQU     0100H   ;TRANSIENT PROGRAM AREA
0900 =                  LOADP   EQU     900H    ;LOAD POINT FOR SYSTEM DURING LOAD/STORE
0005 =                  BDOS    EQU     5H      ;DOS ENTRY POINT
0000 =                  BOOT    EQU     0       ;JMP TO 'BOOT' TO REBOOT SYSTEM
0001 =                  CONI    EQU     1       ;CONSOLE INPUT FUNCTION
0002 =                  CONO    EQU     2       ;CONSOLE OUTPUT FUNCTION
000E =                  SELF    EQU     14      ;SELECT DISK
000F =                  OPENF   EQU     15      ;DISK OPEN FUNCTION
0014 =                  DREADF  EQU     20      ;DISK READ FUNCTION
                        ;
000A =                  MAXTRY  EQU     10      ;MAXIMUM NUMBER OF RETRIES ON EACH READ/WRITE
000D =                  CR      EQU     0DH     ;CARRIAGE RETURN
000A =                  LF      EQU     0AH     ;LINE FEED
0010 =                  STACKSIZE       EQU     16      ;SIZE OF LOCAL STACK
                        ;
0001 =                  WBOOT   EQU     1       ;ADDRESS OF WARM BOOT (OTHER PATCH ENTRY
                        ;                       POINTS ARE COMPUTED RELATIVE TO WBOOT)
0018 =                  SELDSK  EQU     24      ;WBOOT+24 FOR DISK SELECT
001B =                  SETTRK  EQU     27      ;WBOOT+27 FOR SET TRACK FUNCTION
001E =                  SETSEC  EQU     30      ;WBOOT+30 FOR SET SECTOR FUNCTION
0021 =                  SETDMA  EQU     33      ;WBOOT+33 FOR SET DMA ADDRESS
0024 =                  READF   EQU     36      ;WBOOT+36 FOR READ FUNCTION
0027 =                  WRITF   EQU     39      ;WBOOT+39 FOR WRITE FUNCTION
                        ;
0100                            ORG     TPA     ;TRANSIENT PROGRAM AREA
0100 C37802                     JMP     START
0103 434F505952                 DB      'COPYRIGHT (C) 1978, DIGITAL RESEARCH '
                        ;
                        ;       TRANSLATE TABLE - SECTOR NUMBERS ARE TRANSLATED
                        ;       HERE TO DECREASE THE SYSGEN TIME FOR MISSED SECTORS
                        ;       WHEN SLOW CONTROLLERS ARE INVOLVED.  TRANSLATION TAKES
                        ;       PLACE ACCORDING TO THE "SKEW" FACTOR SET ABOVE.
                        ;
0128 02                 OST:    DB      NTRKS   ;OPERATING SYSTEM TRACKS
0129 1A                 SPT:    DB      NSECTS  ;SECTORS PER TRACK (CAN BE PATCHED)
                        TRAN:                   ;BASE OF TRANSLATE TABLE
0001 #                  TRELT   SET     1       ;FIRST/NEXT TRAN ELEMENT
0001 #                  TRBASE  SET     1       ;BASE FOR WRAPAROUND
                                REPT    NSECTS  ;ONCE FOR EACH SECTOR ON A TRACK
                                DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
```

```
                TRELT   SET     TRELT+SKEW
                        IF      TRELT GT NSECTS
                TRBASE  SET     TRBASE+1
                TRELT   SET     TRBASE
                        ENDIF
                        ENDM
012A+01                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
012B+02                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
012C+03                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
012D+04                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
012E+05                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
012F+06                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0130+07                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0131+08                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0132+09                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0133+0A                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0134+0B                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0135+0C                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0136+0D                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0137+0E                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0138+0F                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0139+10                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
013A+11                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
013B+12                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
013C+13                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
013D+14                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
013E+15                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
013F+16                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0140+17                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0141+18                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0142+19                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
0143+1A                 DB      TRELT   ;GENERATE FIRST/NEXT SECTOR
                ;
                ;       NOW LEAVE SPACE FOR EXTENSIONS TO TRANSLATE TABLE
                        IF      NSECTS LT 64
                        REPT    64-NSECTS
                        DB      0
                        ENDM
0144+00                 DB      0
0145+00                 DB      0
0146+00                 DB      0
0147+00                 DB      0
0148+00                 DB      0
0149+00                 DB      0
014A+00                 DB      0
014B+00                 DB      0
014C+00                 DB      0
014D+00                 DB      0
014E+00                 DB      0
014F+00                 DB      0
0150+00                 DB      0
0151+00                 DB      0
0152+00                 DB      0
0153+00                 DB      0
0154+00                 DB      0
0155+00                 DB      0
```

```
0156+00                 DB      0
0157+00                 DB      0
0158+00                 DB      0
0159+00                 DB      0
015A+00                 DB      0
015B+00                 DB      0
015C+00                 DB      0
015D+00                 DB      0
015E+00                 DB      0
015F+00                 DB      0
0160+00                 DB      0
0161+00                 DB      0
0162+00                 DB      0
0163+00                 DB      0
0164+00                 DB      0
0165+00                 DB      0
0166+00                 DB      0
0167+00                 DB      0
0168+00                 DB      0
0169+00                 DB      0
                ;
                ;
                ;
                ;
                ;       UTILITY SUBROUTINES
                MULTSEC:
                        ;MULTIPLY THE SECTOR NUMBER IN A BY THE SECTOR SIZE
016A 6F2600             MOV L,A! MVI H,0 ;SECTOR NUMBER IN HL
                        REPT LOG2SEC    ;LOG 2 OF SECTOR SIZE
                        DAD     H
                        ENDM
016D+29                 DAD     H
016E+29                 DAD     H
016F+29                 DAD     H
0170+29                 DAD     H
0171+29                 DAD     H
0172+29                 DAD     H
0173+29                 DAD     H
0174 C9                 RET ;WITH HL = SECTOR * SECTOR SIZE
                ;
                GETCHAR:
                ;       READ CONSOLE CHARACTER TO REGISTER A
0175 0E01CD0500         MVI C,CONI! CALL BDOS!
                ;       CONVERT TO UPPER CASE BEFORE RETURN
017A FE61D8             CPI 'A' OR 20H ! RC     ;RETURN IF BELOW LOWER CASE A
017D FE7B               CPI ('Z' OR 20H) + 1
017F D0                 RNC     ;RETURN IF ABOVE LOWER CASE Z
0180 E65FC9             ANI 5FH! RET
                ;
                PUTCHAR:
                ;       WRITE CHARACTER FROM A TO CONSOLE
0183 5F0E02CD05         MOV E,A! MVI C,CONO! CALL BDOS! RET
                ;
                CRLF:   ;SEND CARRIAGE RETURN, LINE FEED
018A 3E0D               MVI     A,CR
018C CD8301             CALL    PUTCHAR
```

```
018F 3E0A               MVI     A,LF
0191 CD8301             CALL    PUTCHAR
0194 C9                 RET
                ;
                CRMSG:  ;PRINT MESSAGE ADDRESSED BY H,L TIL ZERO
                        ;WITH LEADING CRLF
0195 E5CD8A01E1         PUSH H! CALL CRLF! POP H ;DROP THRU TO OUTMSGO
                OUTMSG:
019A 7EB7C8             MOV A,M! ORA A! RZ
                ;       MESSAGE NOT YET COMPLETED
019D E5CD8301E1         PUSH H! CALL PUTCHAR! POP H! INX H
01A3 C39A01             JMP     OUTMSG
                ;
                SEL:
                ;       SELECT DISK GIVEN BY REGISTER A
01A6 4F2A010011         MOV C,A! LHLD WBOOT! LXI D,SELDSK! DAD D! PCHL
                ;
                TRK:    ;SET UP TRACK
01AF 2A0100             LHLD    WBOOT   ;ADDRESS OF BOOT ENTRY
01B2 111B00             LXI     D,SETTRK        ;OFFSET FOR SETTRK ENTRY
01B5 19                 DAD     D
01B6 E9                 PCHL            ;GONE TO SETTRK
                ;
                SEC:    ;SET UP SECTOR NUMBER
01B7 2A0100             LHLD    WBOOT
01BA 111E00             LXI     D,SETSEC
01BD 19                 DAD     D
01BE E9                 PCHL
                ;
                DMA:    ;SET DMA ADDRESS TO VALUE OF B,C
01BF 2A0100             LHLD    WBOOT
01C2 112100             LXI     D,SETDMA
01C5 19                 DAD     D
01C6 E9                 PCHL
                ;
                READ:   ;PERFORM READ OPERATION
01C7 2A0100             LHLD    WBOOT
01CA 112400             LXI     D,READF
01CD 19                 DAD     D
01CE E9                 PCHL
                ;
                WRITE:  ;PERFORM WRITE OPERATON
01CF 2A0100             LHLD    WBOOT
01D2 112700             LXI     D,WRITF
01D5 19                 DAD     D
01D6 E9                 PCHL
                ;
                DREAD:  ;DISK READ FUNCTION
01D7 0E14               MVI     C,DREADF
01D9 C30500             JMP     BDOS
                ;
                OPEN:   ;FILE OPEN FUNCTION
01DC 0E0FC30500         MVI C,OPENF ! JMP BDOS
                ;
                GETPUT:
                ;       GET OR PUT CP/M (RW=0 FOR READ, 1 FOR WRITE)
```

```
                ;       DISK IS ALREADY SELECTED
                ;
01E1 210009             LXI     H,LOADP ;LOAD POINT IN RAM FOR CP/M DURING SYSGEN
01E4 22AA04             SHLD    DMADDR
                ;
                ;       CLEAR TRACK TO 00
01E7 3EFF               MVI     A,-1    ;START WITH TRACK EQUAL -1
01E9 32A704             STA     TRACK
                ;
                RWTRK:  ;READ OR WRITE NEXT TRACK
01EC 21A704             LXI     H,TRACK
01EF 34                 INR     M       ;TRACK = TRACK + 1
01F0 3A2801             LDA     OST     ;NUMBER OF OPERATING SYSTEM TRACKS
01F3 BE                 CMP     M       ;= TRACK NUMBER ?
01F4 CA7702             JZ      ENDRW   ;END OF READ OR WRITE
                ;
                ;       OTHERWISE NOTDONE, GO TO NEXT TRACK
01F7 4E                 MOV     C,M     ;TRACK NUMBER
01F8 CDAF01             CALL    TRK     ;TO SET TRACK
01FB 3EFF               MVI     A,-1    ;COUNTS 0, 1, 2, . . . 25
01FD 32A804             STA     SECTOR  ;SECTOR INCREMENTED BEFORE READ OR WRITE
                ;
                RWSEC:  ;READ OR WRITE SECTOR
0200 3A2901             LDA     SPT     ;SECTORS PER TRACK
0203 21A804             LXI     H,SECTOR
0206 34                 INR     M       ;TO NEXT SECTOR
0207 BE                 CMP     M       ;A=26 AND M=0 1 2...25 (USUALLY)
0208 CA6602             JZ      ENDTRK  ;
                ;
                ;       READ OR WRITE SECTOR TO OR FROM CURRENT DMA ADDR
020B 21A804             LXI     H,SECTOR
020E 5E                 MOV     E,M     ;SECTOR NUMBER
020F 1600               MVI     D,0     ;TO DE
0211 212A01             LXI     H,TRAN
0214 46                 MOV     B,M     ;TRAN(0) IN B
0215 19                 DAD     D       ;SECTOR TRANSLATED
0216 4E                 MOV     C,M     ;VALUE TO C READY FOR SELECT
0217 C5                 PUSH    B       ;SAVE TRAN(0),TRAN(SECTOR)
0218 CDB701             CALL    SEC     ;SET UP SECTOR NUMBER
021B C1                 POP     B       ;RECALL TRAN(0),TRAN(SECTOR)
021C 79                 MOV     A,C     ;TRAN(SECTOR)
021D 90                 SUB     B       ;-TRAN(0)
021E CD6A01             CALL    MULTSEC ;*SECTOR SIZE
0221 EB                 XCHG            ;TO DE
0222 2AAA04             LHLD    DMADDR  ;BASE DMA ADDRESS FOR THIS TRACK
0225 19                 DAD     D       ;+(TRAN(SECTOR)-TRAN(0))*SECSIZ
0226 44                 MOV     B,H
0227 4D                 MOV     C,L     ;TO BC FOR SEC CALL
0228 CDBF01             CALL    DMA     ;DMA ADDRESS SET FROM B,C
                ;       DMA ADDRESS SET, CLEAR RETRY COUNT
022B AF                 XRA     A
022C 32AC04             STA     RETRY   ;SET TO ZERO RETRIES
                ;
                TRYSEC: ;TRY TO READ OR WRITE CURRENT SECTOR
022F 3AAC04             LDA     RETRY
0232 FE0A               CPI     MAXTRY  ;TOO MANY RETRIES?
```

```
0234 DA4B02             JC      TRYOK
                ;
                ;       PAST MAXTRIES, MESSAGE AND IGNORE
0237 211804             LXI     H,ERRMSG
023A CD9A01             CALL    OUTMSG
023D CD7501             CALL    GETCHAR
0240 FE0D               CPI     CR
0242 C26303             JNZ     REBOOT
                ;
                ;       TYPED A CR, OK TO IGNORE
0245 CD8A01             CALL    CRLF
0248 C30002             JMP     RWSEC
                ;
                TRYOK:
                ;       OK TO TRY READ OR WRITE
024B 3C                 INR     A
024C 32AC04             STA     RETRY   ;RETRY=RETRY+1
024F 3AA904             LDA     RW      ;READ OR WRITE?
0252 B7                 ORA     A
0253 CA5C02             JZ      TRYREAD
                ;
                ;       MUST BE WRITE
0256 CDCF01             CALL    WRITE
0259 C35F02             JMP     CHKRW   ;CHECK FOR ERROR RETURNS
                TRYREAD:
025C CDC701             CALL    READ
                CHKRW:
025F B7                 ORA     A
0260 CA0002             JZ      RWSEC   ;ZERO FLAG IF R/W OK
                ;
                ;       ERROR, RETRY OPERATION
0263 C32F02             JMP     TRYSEC
                ;
                ;       END OF TRACK
                ENDTRK:
0266 3A2901             LDA     SPT     ;SECTORS PER TRACK
0269 CD6A01             CALL    MULTSEC ;*SECSIZ
026C EB                 XCHG            ;TO DE
026D 2AAA04             LHLD    DMADDR  ;BASE DMA FOR THIS TRACK
0270 19                 DAD     D       ;+SPT*SECSIZ
0271 22AA04             SHLD    DMADDR  ;READY FOR NEXT TRACK
0274 C3EC01             JMP     RWTRK   ;FOR ANOTHER TRACK
                ;
                ENDRW:  ;END OF READ OR WRITE, RETURN TO CALLER
0277 C9                 RET
                ;
                ;
                START:
                ;
0278 31CD04             LXI     SP,STACK        ;SET LOCAL STACK POINTER
027B 217503             LXI     H,SIGNON
027E CD9A01             CALL    OUTMSG
                ;
                ;       CHECK FOR DEFAULT FILE LOAD INSTEAD OF GET
                ;
0281 3A5D00             LDA     FCB+1   ;BLANK IF NO FILE
```

```
0284 FE20               CPI     ' '
0286 CAD602             JZ      GETSYS  ;SKIP TO GET SYSTEM MESSAGE IF BLANK
0289 115C00             LXI     D,FCB   ;TRY TO OPEN IT
028C CDDC01             CALL    OPEN    ;
028F 3C                 INR     A       ;255 BECOMES 00
0290 C29C02             JNZ     RDOK    ;OK TO READ IF NOT 255
                ;
                ;       FILE NOT PRESENT, ERROR AND REBOOT
                ;
0293 21780"             LXI     H,NOFILE
0296 CD9501             CALL    CRMSG
0299 C36303             JMP     REBOOT
                ;
                ;       FILE PRESENT
                ;         READ TO LOAD POINT
                ;
                RDOK:
029C AF                 XRA     A
029D 327C00             STA     FCBCR   ;CURRENT RECORD = 0
                ;
                ;       PRE-READ AREA FROM TPA TO LOADP
                ;
02A0 0E10               MVI     C,(LOADP-TPA)/SECSIZ
                ;       PRE-READ FILE
                PRERD:
02A2 C5                 PUSH    B       ;SAVE COUNT
02A3 115C00             LXI     D,FCB   ;INPUT FILE CONTROL COUNT
02A6 CDD701             CALL    DREAD   ;ASSUME SET TO DEFAULT BUFFER
02A9 C1                 POP     B       ;RESTORE COUNT
02AA B7                 ORA     A
02AB C2CD02             JNZ     BADRD   ;CANNOT ENCOUNTER END-OF FILE
02AE 0D                 DCR     C       ;COUNT DOWN
02AF C2A202             JNZ     PRERD   ;FOR ANOTHER SECTOR
                ;
                ;       SECTORS SKIPPED AT BEGINNING OF FILE
                ;
02B2 210009             LXI     H,LOADP
                RDINP:
02B5 E5                 PUSH    H
02B6 44                 MOV     B,H
02B7 4D                 MOV     C,L     ;READY FOR DMA
02B8 CDBF01             CALL    DMA     ;DMA ADDRESS SET
02BB 115C00             LXI     D,FCB   ;READY FOR READ
02BE CDD701             CALL    DREAD   ;
02C1 E1                 POP     H       ;RECALL DMA ADDRESS
02C2 B7                 ORA     A       ;00 IF READ OK
02C3 C21C03             JNZ     PUTSYS  ;ASSUME EOF IF NOT.
                ;       MORE TO READ, CONTINUE
02C6 118000             LXI     D,SECSIZ
02C9 19                 DAD     D       ;HL IS NEW LOAD ADDRESS
02CA C3B502             JMP     RDINP
                ;
                BADRD:  ;EOF ENCOUNTERED IN INPUT FILE

02CD 218F04             LXI     H,BADFILE
02D0 CD9501             CALL    CRMSG
```

```
02D3 C36303             JMP     REBOOT
                ;
                ;
                GETSYS:
02D6 218403             LXI     H,ASKGET        ;GET SYSTEM?
02D9 CD9501             CALL    CRMSG
02DC CD7501             CALL    GETCHAR
02DF FE0D               CPI     CR
02E1 CA1C03             JZ      PUTSYS  ;SKIP IF CR ONLY
                ;
02E4 D641               SUI     'A'             ;NORMALIZE DRIVE NUMBER
02E6 FE04               CPI     NDISKS          ;VALID DRIVE?
02E8 DAF102             JC      GETC            ;SKIP TO GETC IF SO
                ;
                ;       INVALID DRIVE NUMBER
02EB CD6E03             CALL    BADDISK
02EE C3D602             JMP     GETSYS          ;TO TRY AGAIN
                ;
                GETC:
                ;       SELECT DISK GIVEN BY REGISTER A
02F1 C641               ADI     'A'
02F3 32B403             STA     GDISK           ;TO SET MESSAGE
02F6 D641               SUI     'A'
02F8 CDA601             CALL    SEL             ;TO SELECT THE DRIVE
                ;       GETSYS, SET RW TO READ AND GET THE SYSTEM
02FB CD8A01             CALL    CRLF
02FE 21AA03             LXI     H,GETMSG
0301 CD9A01             CALL    OUTMSG
0304 CD7501             CALL    GETCHAR
0307 FE0D               CPI     CR
0309 C26303             JNZ     REBOOT
030C CD8A01             CALL    CRLF
                ;
030F AF                 XRA     A
0310 32A904             STA     RW
0313 CDE101             CALL    GETPUT
0316 213F04             LXI     H,DONE
0319 CD9A01             CALL    OUTMSG
                ;
                ;       PUT SYSTEM
                PUTSYS:
031C 21C803             LXI     H,ASKPUT
031F CD9501             CALL    CRMSG
0322 CD7501             CALL    GETCHAR
0325 FE0D               CPI     CR
0327 CA6303             JZ      REBOOT
032A D641               SUI     'A'
032C FE04               CPI     NDISKS
032E DA3703             JC      PUTC
                ;
                ;       INVALID DRIVE NAME
0331 CD6E03             CALL    BADDISK
0334 C31C03             JMP     PUTSYS  ;TO TRY AGAIN
                ;
                PUTC:
                ;       SET DISK FROM REGISTER C
```

```
0337 C641              ADI     'A'
0339 320404            STA     PDISK   ;MESSAGE SET
033C D641              SUI     'A'
033E CDA601            CALL    SEL     ;SELECT DEST DRIVE
                ;      PUT SYSTEM, SET RW TO WRITE
0341 21F503            LXI     H,PUTMSG
0344 CD9501            CALL    CRMSG
0347 CD7501            CALL    GETCHAR
034A FE0D              CPI     CR
034C C26303            JNZ     REBOOT
034F CD8A01            CALL    CRLF
                ;
0352 21A904            LXI     H,RW
0355 3601              MVI     M,1
0357 CDE101            CALL    GETPUT  ;TO PUT SYSTEM BACK ON DISKETTE
035A 213F04            LXI     H,DONE
035D CD9A01            CALL    OUTMSG
0360 C31C03            JMP     PUTSYS  ;FOR ANOTHER PUT OPERATION
                ;
                REBOOT:
0363 3E00              MVI     A,0
0365 CDA601            CALL    SEL
0368 CD8A01            CALL    CRLF
036B C30000            JMP     BOOT
                BADDISK:
                       ;BAD DISK NAME
036E 215104            LXI     H,QDISK
0371 CD9501            CALL    CRMSG
0374 C9                RET
                ;
                ;
                ;
                ;      DATA AREAS
                ;      MESSAGES
0375 5359534745SIGNON: DB      'SYSGEN VER '
0380 322E30            DB      VERS/10+'0','.',VERS MOD 10+'0'
0383 00                DB      0
0384 534F555243ASKGET: DB      'SOURCE DRIVE NAME (OR RETURN TO SKIP)',0
03AA 534F555243GETMSG: DB      'SOURCE ON '
03B4           GDISK:  DS      1       ;FILLED IN AT GET FUNCTION
03B5 2C20544845        DB      ', THEN TYPE RETURN',0
03C8 4445535449ASKPUT: DB      'DESTINATION DRIVE NAME (OR RETURN TO REBOOT)',0
03F5 4445535449PUTMSG: DB      'DESTINATION ON '
0404           PDISK:  DS      1       ;FILLED IN AT PUT FUNCTION
0405 2C20544845        DB      ', THEN TYPE RETURN',0
0418 5045524D41ERRMSG: DB      'PERMANENT ERROR, TYPE RETURN TO IGNORE',0
043F 46554E4354DONE:   DB      'FUNCTION COMPLETE',0
0451 494E56414CQDISK:  DB      'INVALID DRIVE NAME (USE A, B, C, OR D)',0
0478 4E4F20534FNOFILE: DB      'NO SOURCE FILE ON DISK',0
                BADFILE:
048F 534F555243        DB      'SOURCE FILE INCOMPLETE',0
                ;
                ;      VARIABLES
04A6           SDISK:  DS      1       ;SELECTED DISK FOR CURRENT OPERATION
04A7           TRACK:  DS      1       ;CURRENT TRACK
04A8           SECTOR: DS      1       ;CURRENT SECTOR
```

```
04A9            RW:     DS      1       ;READ IF 0, WRITE IF 1
04AA            DMADDR: DS      2       ;CURRENT DMA ADDRESS
04AC            RETRY:  DS      1       ;NUMBER OF TRIES ON THIS SECTOR
04AD                    DS      STACKSIZE*2
                STACK:
B04CD                   END
```

0384 ASKGET	03C8 ASKPUT	036E BADDISK	048F BADFILE	02CD BADRD
0005 BDOS	0000 BOOT	025F CHKRW	0001 CONI	0002 CONO
000D CR	018A CRLF	0195 CRMSG	01BF DMA	04AA DMADDR
043F DONE	0014 DREADF	01D7 DREAD	0277 ENDRW	0266 ENDTRK
0418 ERRMSG	005C FCB	007C FCBCR	03B4 GDISK	02F1 GETC
0175 GETCHAR	03AA GETMSG	01E1 GETPUT	02D6 GETSYS	000A LF
0900 LOADP	0007 LOG2SEC	000A MAXTRY	016A MULTSEC	0004 NDISKS
0478 NOFILE	001A NSECTS	0002 NTRKS	01DC OPEN	000F OPENF
0128 OST	019A OUTMSG	0404 PDISK	02A2 PRERD	0183 PUTCHAR
0337 PUTC	03F5 PUTMSG	031C PUTSYS	0451 QDISK	02B5 RDINP
029C RDOK	0024 READF	01C7 READ	0363 REBOOT	04AC RETRY
04A9 RW	0200 RWSEC	01EC RWTRK	04A6 SDISK	01B7 SEC
0080 SECSIZ	04A8 SECTOR	0018 SELDSK	000E SELF	01A6 SEL
0021 SETDMA	001E SETSEC	001B SETTRK	0375 SIGNON	0001 SKEW
0129 SPT	04CD STACK	0010 STACKSIZE	0278 START	0100 TPA
04A7 TRACK	012A TRAN	01AF TRK	024B TRYOK	025C TRYRE'D
022F TRYSEC	0014 VERS	0001 WBOOT	01CF WRITE	0027 WRITF