```
ASSMX 0

0000                    0005 ************************************************
0000                    0010 *  EXATRON STRINGY FLOPPY UTILITY ROUTINES  *
0000                    0015 *  VERSION 3.2                    07-02-79  *
0000                    0020 ************************************************
0000                    0025 *
0000                    0030 * WRITTEN BY:  CHARLIE PACK
0000                    0035 *              25470 ELENA ROAD
0000                    0040 *              LOS ALTOS HILLS, CA.  94022
0000                    0045 *              (415) 941-0495  EVENINGS
0000                    0050 *
0000                    0055 * CHANGES FROM VERSION 3.1 ARE:
0000                    0060 *  1. IPL ROUTINE GETS STACK PAGE FROM INTERNAL
0000                    0065 *     DIPSWITCH INSTEAD OF FRONT PANEL SENSE SW.
0000                    0070 *     RETURN CODE 07 NOW STORED IN FCB (BUG FIX).
0000                    0075 *  2. SINGLE DENSITY MODE IS FORCED FOR ALL OP-
0000                    0080 *     ERATIONS.
0000                    0085 *  3. IF A TAPE HAS A CAPACITY OF >= HEX OFFFFH
0000                    0090 *     BYTES, HEX OFFFEH BYTES WILL BE CERTIFIED.
0000                    0095 *     IN 3.1 THIS DIDN'T WORK PROPERLY.
0000                    0100 *  4. COMMENT CORRECTIONS IN CERTIFY PROCEDURE.
0000                    0105 *  5. LAST 4 BYTES NO LONGER RESERVED FOR DATA
0000                    0110 *     AND CONTROL PORTS.  THIS IS ONLY REQUIRED
0000                    0115 *     FOR MODULE OCCUPYING LAST 1K BLOCK.
0000                    0120 *  6. STILL MORE CODE COMPRESSION!
0000                    0125 *
0000                    0130 * STATUS MASKS
0000                    0135 *
0000                    0140 TBEM   EQU  01H ;TRANSMITTER BUFFER EMPTY
0000                    0145 RDYM   EQU  02H ;DATA READY
0000                    0150 PAEM   EQU  08H ;PARITY ERROR
0000                    0155 EOTM   EQU  80H ;END OF TAPE
0000                    0160 *
0000                    0165 * USART CONTROL WORDS AND COMMANDS
0000                    0170 *
0000                    0175 CCHAR  EQU  094H ;CERTIFY TEST CHARACTER
0000                    0180 CMARK  EQU  0CCH ;CERTIFY FILE MARK
0000                    0185 FMARK  EQU  055H ;DATA FILE MARK
0000                    0190 SYNC   EQU  016H ;SYNC WORD
0000                    0195 *
0000                    0200 MOFC   EQU  031H ;MOTOR-OFF
0000                    0205 MTRC   EQU  012H ;MOTOR-ON
0000                    0210 RCVC   EQU  016H ;RECEIVER-ON
0000                    0215 XMTC   EQU  033H ;TRANSMIT
0000                    0220 RSTC   EQU  040H ;RE-SET
0000                    0225 HUNC   EQU  096H ;HUNT
0000                    0230 MODC   EQU  0BCH ;MODE
0000
```

```
                    0235 *
0000                0240 * RETURN CODES
0000                0245 *
0000                0250 RC0    EQU  00H ;NORMAL (GOOD) RETURN CODE
0000                0255 RC1    EQU  01H ;UNCORRECTABLE P.E.
0000                0260 RC2    EQU  02H ;FILE NOT FOUND
0000                0265 RC3    EQU  03H ;VERIFY ERROR
0000                0270 RC4    EQU  04H ;TAPE IS FULL
0000                0275 RC5    EQU  05H ;WRITE PROTECTED
0000                0280 RC6    EQU  06H ;FILE POSITION ERROR
0000                0285 RC7    EQU  07H ;CAN'T AUTOSTART
0000                0290 RC8    EQU  08H ;BAD MEMORY
0000                0295 *
0000                0300 * OTHER EQUATES
0000                0305 *
0000                0310 PSW    EQU  6
0000                0315 SP     EQU  6
0000                0320 RAMCH  EQU  06DH ;RAM CHECK CHARACTER
0000                0325 LEADL  EQU  00140H ;BYTE LENGTH OF LEADER
0000                0330 IBGPL  EQU  00140H ;LENGTH OF INTER-BLOCK GAP
0000                0335 TRLRL  EQU  0000AH ;BYTE LENGTH OF TRAILER
0000                0340 START  EQU  $
0000                0345 ESFCT  EQU  START ;ADDR OF ESF CONTROLLER
0000                0350 CTLPT  EQU  ESFCT+0FFFH ;CONTROL PORT
0000                0355 DATPT  EQU  ESFCT+0FFEH ;DATA PORT
0000                0360 SWTPT  EQU  ESFCT+0FFBH ;DIPSWITCH PORT
0000
```

```
                        0365 *
                        0370 * ESF UTILITY SYSTEM ENTRY POINTS
0000                    0375 *
0000 F3                 0380 EIPL  DI          ;PROGRAM IDENTIFIER
0001 C3 1D 00           0385       JMP  IPL    ;INITIAL PROGRAM LOAD
0004                    0390 *
0004 F3                 0395 EMON  DI          ;PROGRAM IDENTIFIER
0005 C3 00 04           0400       JMP  START+400H ;JUMP TO MONITOR
0008                    0405 *
0008 C3 6B 00           0410 ECAT  JMP  CAT    ;GET SYSTEM CATALOG
000B C3 98 00           0415 EGET  JMP  GET    ;GET OR VERIFY A DATA BLOCK
000E C3 5E 01           0420 ESAV  JMP  SAVE   ;SAVE A DATA BLOCK
0011 C3 40 02           0425 ECER  JMP  CERT   ;CERTIFY A TAPE
0014                    0430 EWTP  EQU  $      ;WIND TAPE
0014                    0435 *
0014                    0440 * WIND TAPE - POSITIONS BOT MARKER
0014                    0445 * AT EXPOSED POSITION
0014                    0450 *
0014 CD FA 02           0455       CALL INITS ;INITIALIZE CONTROLLER
0017 CD 4E 03           0460       CALL BOT2  ;FIND BEGINNING OF TAPE
001A C3 F6 03           0465       JMP  RESET ;RESET CONTROLLER & RETURN
001D                    0470 *
001D                    0475 * INITIAL PROGRAM LOAD
001D                    0480 *
001D 3A FB 0F           0485 IPL   LDA  SWTPT ;READ DIPSWITCH PORT
0020 2F                 0490       CMA        ;INVERT IT
0021 E6 0F              0495       ANI  0FH   ;MASK RIGHT NYBBLE
0023 07                 0500       RLC        ;SHIFT LEFT 4 BITS
0024 07                 0505       RLC
0025 07                 0510       RLC
0026 07                 0515       RLC
0027 F6 0F              0520       ORI  0FH   ;MASK TO LAST PAGE OF 4K BLK
0029 67                 0525       MOV  H,A
002A 2E F3              0530       MVI  L,0FFH-FCBLN+FCB+1
002C                    0535 * HL NOW HAS ADDRESS OF TOP OF STACK + 1,
002C                    0540 * WHICH IS ALSO ADDRESS OF FCB.  FCB IS AT
002C                    0545 * TOP OF CHOSEN RAM PAGE AND STACK IS JUST
002C                    0550 * BELOW FCB.
002C F9                 0555       SPHL       ;LOAD STACK POINTER
002D 3E 6D              0560       MVI  A,RAMCH
002F 77                 0565       MOV  M,A   ;CHECK FOR PRESENCE
0030 BE                 0570       CMP  M     ;  OF READ/WRITE MEMORY
0031 3E 08              0575       MVI  A,RC8 ;ERROR RETURN CODE
0033 C2 65 00           0580       JNZ  IPLER ;BRANCH IF BAD MEMORY
0036                    0585 *
0036 E5                 0590       PUSH H
0037 AF                 0595       XRA  A     ;ACC.=0
0038 77                 0600       MOV  M,A   ;RETURN CODE TO FCB
0039 23                 0605       INX  H     ;HL=&FCBOP
003A 36 58              0610       MVI  M,'X' ;OPTION (AUTOEXECUTE) TO FCB
003C 23                 0615       INX  H
003D 77                 0620       MOV  M,A   ;FILE POS. (0) TO FCB
003E 23                 0625       INX  H     ;HL=&FCBID
003F 77                 0630       MOV  M,A   ;FILE ID TO FCB
0040 23                 0635       INX  H     ;HL=&FCBTY
0041 36 41              0640       MVI  M,'A' ;FILE TYPE TO FCB
0043 23                 0645       INX  H     ;HL=&FCBBG
0044 3D                 0650       DCR  A     ;ACC.=0FFH
0045 77                 0655       MOV  M,A
0046 23                 0660       INX  H
0047 77                 0665       MOV  M,A   ;DUMMY BEGIN. ADDR TO FCB
0048 E1                 0670       POP  H     ;HL=&FCB
0049
```

```
                        0675 *
0049 CD 98 00           0680        CALL GET    ;GET PROGRAM INTO MEMORY
004C DA 65 00           0685        JC   IPLER  ;JUMP IF ERROR
004F                    0690 *
004F 47                 0695        MOV  B,A    ;SAVE RETURN CODE
0050 3E 09              0700        MVI  A,FCBAS-FCB
0052 CD E2 02           0705        CALL EXT    ;GET AUTOSTART
0055 EB                 0710        XCHG        ;ADDR IN HL
0056 23                 0715        INX  H
0057 7C                 0720        MOV  A,H    ;IS AUTOSTART IMPLE-
0058 B5                 0725        ORA  L      ;MENTED ON THIS FILE?
0059 2B                 0730        DCX  H
005A CA 62 00           0735        JZ   NOAS   ;NO, CANNOT AUTOSTART.
005D 78                 0740        MOV  A,B    ;RESTORE RETURN CODE
005E 2F                 0745        CMA         ;SHOW RETURN CODE IN DATA
005F D3 FF              0750        OUT  OFFH   ;LIGHTS OF IMSAI 8080
0061 E9                 0755        PCHL        ;AND JUMP TO PROGRAM.
0062                    0760 *
0062 3E 07              0765 NOAS   MVI  A,RC7  ;ERROR RETURN CODE
0064 12                 0770        STAX D      ;  TO FCB
0065 2F                 0775 IPLER  CMA         ;SHOW RETURN CODE IN DATA
0066 D3 FF              0780        OUT  OFFH   ;LIGHTS OF IMSAI 8080
0068 C3 68 00           0785 HALT   JMP  HALT   ;AND SOFT-HALT.
006B
```

```
                        0790 *
006B                    0795 * MAIN PROCEDURE TO EXAMINE THE CATALOG OF
006B                    0800 * EACH FILE
006B                    0805 * ENTRY: HL = ADDRESS OF FILE CONTROL BLOCK
006B                    0810 * EXIT: ZERO & CARRY FLAGS OFF = FILE CONTROL
006B                    0815 *           BLOCK HAS CATALOG OF NEXT FILE.
006B                    0820 *           IF FCBOP WAS ZERO AT ENTRY, THIS
006B                    0825 *           WILL BE THE CATALOG OF THE FIRST
006B                    0830 *           FILE.
006B                    0835 *         ZERO FLAG SET = END OF TAPE, NO
006B                    0840 *           MORE DATA (NOT AN ERROR)
006B                    0845 *         CARRY FLAG SET = ERROR FOUND ON
006B                    0850 *           THIS FILE, FCBRC HAS NON-ZERO
006B                    0855 *           RETURN CODE
006B                    0860 *
006B E5                 0865 CAT    PUSH H      ;(SP)=&FCB
006C 23                 0870        INX  H      ;HL=&FCBOP
006D 36 43              0875        MVI  M,'C' ;OPTION TO FCB
006F                    0880 *
006F CD FA 02           0885        CALL INITS ;INITIALIZE CONTROLLER
0072                    0890 *
0072 E1                 0895        POP  H      ;HL=&FCB
0073 E5                 0900        PUSH H
0074 23                 0905        INX  H
0075 23                 0910        INX  H      ;HL=&FCBFP
0076 7E                 0915        MOV  A,M    ;ACC=PREVIOUS FILE POS.
0077 34                 0920        INR  M      ;BUMP CURRENT FILE POS.
0078                    0925 *
0078 B7                 0930        ORA  A      ;FIRST TIME THROUGH?
0079 CC 4B 03           0935        CZ   BOT1   ;IF SO, FIND BEGIN-
007C                    0940 *                  ;NING OF TAPE.
007C 0E 55              0945        MVI  C,FMARK
007E CD 5B 03           0950        CALL GETFM ;GET NEXT FILE MARK
0081 CA 33 02           0955        JZ   ENDTP ;JUMP IF EOT
0084                    0960 *
0084 CD 75 03           0965        CALL GETWD ;GET FILE ID/TYPE IN D-E
0087 5F                 0970        MOV  E,A
0088 CD 75 03           0975        CALL GETWD
008B 57                 0980        MOV  D,A
008C E1                 0985        POP  H      ;HL=&FCB
008D 3E 03              0990        MVI  A,FCBID-FCB
008F CD EE 02           0995        CALL INS    ;FILE ID/TYPE TO FCB
0092 11 FF FF           1000        LXI  D,0FFFFH ;DUMMY LOAD ADDR
0095 C3 AF 00           1005        JMP  GET2   ;GO TO "GET" ROUTINE
0098
```

```
                        1010 *
0098                    1015 * MAIN PROCEDURE TO GET A FILE FROM TAPE.
0098                    1020 * ENTRY: HL = ADDRESS OF FILE CONTROL BLOCK
0098                    1025 * EXIT: CARRY FLAG OFF = SUCCESSFUL OPERATION
0098                    1030 *       CARRY FLAG SET = NON-ZERO RETURN CODE
0098                    1035 *          IN ACC. AND IN FCBRC; DATA ERROR
0098                    1040 *          ADDR IN FCBER.
0098                    1045 *
0098 E5                 1050 GET    PUSH H      ;(SP)=&FCB
0099 3E 03              1055        MVI  A,FCBID-FCB
009B CD E2 02           1060        CALL EXT    ;GET FILE ID/TYPE FROM FCB
009E                    1065 *                  ;D=FILE TYPE, E=FILE ID
009E CD FA 02           1070        CALL INITS  ;INITIALIZE CONTROLLER
00A1 0E 55              1075        MVI  C,FMARK
00A3 CD 1F 03           1080        CALL FIND   ;FIND THE REQUIRED FILE
00A6 DA 18 02           1085        JC   NOFIL  ;JUMP IF NOT FOUND
00A9 E1                 1090        POP  H      ;&FCB
00AA                    1095 *
00AA                    1100 * OBTAIN LOAD ADDRESS AND COUNT.  THE LOAD
00AA                    1105 * ADDRESS IN FCBBG OVERRIDES THE LOAD AD-
00AA                    1110 * DRESS ON THE TAPE, UNLESS FCBBG CONTAINS
00AA                    1115 * THE DUMMY ADDRESS OFFFFH.
00AA                    1120 *
00AA 3E 05              1125        MVI  A,FCBBG-FCB
00AC CD E2 02           1130        CALL EXT    ;GET FCBBG INTO D-E
00AF E5                 1135 GET2   PUSH H
00B0                    1140 *
00B0 AF                 1145        XRA  A
00B1 47                 1150        MOV  B,A    ;RE-SET PARITY ERROR COUNT
00B2 4F                 1155        MOV  C,A    ;AND LONG. PARITY VALUE
00B3                    1160 *
00B3 EB                 1165        XCHG
00B4 CD 8A 03           1170        CALL GETDW  ;GET LOAD ADDR IN D-E
00B7 DA 13 02           1175        JC   UNCPE  ;JUMP IF PARITY ERROR
00BA                    1180 *                  ;DE=LOAD ADDR FROM FILE
00BA                    1185 *                  ;HL=LOAD ADDR FROM FCB
00BA 23                 1190        INX  H
00BB 7C                 1195        MOV  A,H
00BC B5                 1200        ORA  L      ;HL=DUMMY LOAD ADDR?
00BD 2B                 1205        DCX  H
00BE C2 C9 00           1210        JNZ  GET3   ;NO, USE IT
00C1                    1215 *
00C1 E1                 1220        POP  H      ;HL=&FCB
00C2 3E 05              1225        MVI  A,FCBBG-FCB
00C4 CD EE 02           1230        CALL INS    ;INSERT LOAD ADDR IN FCB
00C7 E5                 1235        PUSH H
00C8 EB                 1240        XCHG        ;HL=THE REAL LOAD ADDR
00C9                    1245 *
00C9 CD 8A 03           1250 GET3   CALL GETDW  ;GET COUNT IN D-E
00CC DA 13 02           1255        JC   UNCPE  ;JUMP IF PARITY ERROR
00CF
```

```
                        1260 *
00CF                    1265 * VERIFY OR GET DATA BLOCK INTO MEMORY
00CF                    1270 *
00CF CD 94 03           1275 GETNX CALL GETCH ;GET A DATA CHARACTER
00D2 EB                 1280       XCHG       ;DE=MEMORY POINTER
00D3 E3                 1285       XTHL       ;HL=&FCB, (SP)=COUNT
00D4 C5                 1290       PUSH B     ;SAVE ERROR DATA
00D5 47                 1295       MOV  B,A   ;B=DATA FROM TAPE
00D6                    1300 *
00D6 D2 DE 00           1305       JNC  GET4  ;JUMP IF NO ERROR
00D9 3E 0B              1310       MVI  A,FCBER-FCB
00DB CD EE 02           1315       CALL INS   ;ELSE, INSERT ADDR OF
00DE                    1320 *                ;ERROR DATA IN FCB.
00DE                    1325 *
00DE 23                 1330 GET4  INX  H     ;HL=&OPTION CODE
00DF 7E                 1335       MOV  A,M   ;ACC=OPTION (C,E,V,X)
00E0 2B                 1340       DCX  H     ;HL=&FCB
00E1                    1345 *
00E1 FE 56              1350       CPI  'V'   ;VERIFY ONLY?
00E3 CA F0 00           1355       JZ   VERFY ;YES
00E6 FE 43              1360       CPI  'C'   ;GET CATALOG ONLY?
00E8 CA F5 00           1365       JZ   GET5  ;YES
00EB                    1370 *
00EB 78                 1375       MOV  A,B
00EC 12                 1380       STAX D     ;STORE DATA IN MEMORY
00ED C3 F5 00           1385       JMP  GET5
00F0                    1390 *
00F0 1A                 1395 VERFY LDAX D     ;ACC. HAS CORRECT DATA
00F1 B8                 1400       CMP  B     ;SAME AS DATA FROM TAPE?
00F2 C2 53 01           1405       JNZ  FILER ;NO
00F5                    1410 *
00F5 C1                 1415 GET5  POP  B     ;RESTORE ERROR DATA
00F6 E3                 1420       XTHL       ;(SP)=&FCB
00F7 EB                 1425       XCHG       ;HL=MEMORY POINTER
00F8                    1430 *                ;DE=COUNT
00F8 23                 1435       INX  H     ;BUMP MEMORY POINTER
00F9 1B                 1440       DCX  D     ;DECR. COUNT
00FA 7A                 1445       MOV  A,D
00FB B3                 1450       ORA  E     ;IS COUNT = 0 YET?
00FC C2 CF 00           1455       JNZ  GETNX ;NO, CONTINUE LOOPING
00FF                    1460 *
00FF 2B                 1465       DCX  H
0100 EB                 1470       XCHG       ;DE=ENDING ADDR
0101 E1                 1475       POP  H     ;HL=&FCB
0102 3E 07              1480       MVI  A,FCBEN-FCB
0104 CD EE 02           1485       CALL INS   ;PUT ENDING ADDR IN FCB
0107 E5                 1490       PUSH H     ;(SP)=&FCB
0108                    1495 *
0108 CD 8A 03           1500       CALL GETDW ;GET AUTOSTART IN D-E
010B DA 13 02           1505       JC   UNCPE ;JUMP IF PARITY ERROR
010E E1                 1510       POP  H     ;&FCB
010F 3E 09              1515       MVI  A,FCBAS-FCB
0111 CD EE 02           1520       CALL INS   ;PUT AUTOSTART IN FCB
0114 E5                 1525       PUSH H
0115
```

```
                        1530 *
0115 51                 1535        MOV  D,C    ;D=COMPUTED LONG. PARITY
0116 78                 1540        MOV  A,B
0117 FE 01              1545        CPI  1      ;CHECK PARITY ERR COUNT
0119 CA 29 01           1550        JZ   ONEPE  ;JUMP IF ONE P.E.
011C D2 13 02           1555        JNC  UNCPE  ;JUMP IF >1 P.E.
011F                    1560 *
011F CD 94 03           1565        CALL GETCH  ;GET LONG. PARITY
0122 92                 1570        SUB  D      ;SAME AS COMPUTED?
0123 C2 13 02           1575        JNZ  UNCPE  ;NO, NOT CORRECTABLE
0126 C3 37 02           1580        JMP  DONE   ;YES, WE ARE DONE
0129                    1585 *
0129                    1590 * CORRECT SINGLE PARITY ERROR
0129                    1595 *
0129 CD 94 03           1600 ONEPE  CALL GETCH  ;GET LONG. PARITY
012C BA                 1605        CMP  D      ;SAME AS COMPUTED?
012D CA 13 02           1610        JZ   UNCPE  ;YES, NOT CORRECTABLE
0130 AA                 1615        XRA  D      ;MASK UNMATCHED BITS
0131 47                 1620        MOV  B,A    ;B=UNMATCHED BITS
0132 17                 1625 ROLL   RAL         ;FIND UNMATCHED BIT
0133 D2 32 01           1630        JNC  ROLL
0136 B7                 1635        ORA  A      ;>1 UNMATCHED BITS?
0137 C2 13 02           1640        JNZ  UNCPE  ;YES, NOT CORRECTABLE
013A                    1645 *
013A 23                 1650        INX  H      ;HL=&OPTION CODE
013B 7E                 1655        MOV  A,M    ;ACC.=OPTION CODE
013C 2B                 1660        DCX  H      ;HL=&FCB
013D FE 43              1665        CPI  'C'    ;CATALOG OPTION?
013F CA 37 02           1670        JZ   DONE   ;YES, DON'T CORRECT ERR.
0142 FE 56              1675        CPI  'V'    ;VERIFY ONLY OPTION?
0144 CA 37 02           1680        JZ   DONE   ;YES
0147                    1685 *
0147 3E 0B              1690        MVI  A,FCBER-FCB
0149 CD E2 02           1695        CALL EXT    ;GET ERROR
014C EB                 1700        XCHG        ;DATA ADDR IN HL
014D 7E                 1705        MOV  A,M    ;ACC. HAS BAD BYTE
014E A8                 1710        XRA  B      ;COMPLEMENT BAD BIT
014F 77                 1715        MOV  M,A    ;AND SAVE IT IN MEMORY
0150 C3 37 02           1720        JMP  DONE
0153                    1725 *
0153                    1730 * SET RETURN CODE FOR FILE VERIFY ERROR
0153                    1735 *
0153 3E 0B              1740 FILER  MVI  A,FCBER-FCB
0155 CD EE 02           1745        CALL INS    ;INSERT ADDR OF
0158                    1750 *                  ;ERROR DATA IN FCB
0158 C1                 1755        POP  B
0159 E3                 1760        XTHL        ;(SP)=&FCB
015A EB                 1765        XCHG        ;HL=MEMORY POINTER
015B                    1770 *                  ;DE=COUNT
015B C3 1D 02           1775        JMP  VFYER
015E
```

```
                        1780 *
015E                    1785 * MAIN PROCEDURE TO SAVE A FILE ON TAPE.
015E                    1790 * ENTRY: HL = ADDRESS OF FILE CONTROL BLOCK
015E                    1795 * EXIT: CARRY FLAG OFF = SUCCESSFUL OPERATION
015E                    1800 *       CARRY FLAG SET = NON-ZERO RETURN CODE
015E                    1805 *          IN ACC. AND IN FCBRC.
015E                    1810 *
015E E5                 1815 SAVE  PUSH H      ;(SP)=&FCB
015F CD FA 02           1820       CALL INITS  ;INITIALIZE CONTROLLER
0162                    1825 *
0162 E1                 1830       POP  H      ;&FCB
0163 E5                 1835       PUSH H
0164 23                 1840       INX  H
0165 23                 1845       INX  H      ;HL=&FCBFP
0166 11 40 01           1850       LXI  D,IBGPL ;DE=LENGTH OF GAP
0169 7E                 1855       MOV  A,M    ;ACC=FILE POSITION
016A B7                 1860       ORA  A      ;IS FILE POSITION = 0?
016B CA 9F 01           1865       JZ   SAVE3  ;YES, WRITE FILE FROM
016E                    1870 *                 ;CURRENT POSITION OF TAPE.
016E 47                 1875       MOV  B,A    ;B=FILE POSITION
016F CD 4B 03           1880       CALL BOT1   ;FIND BEGINNING OF TAPE
0172 05                 1885       DCR  B      ;FIRST FILE?
0173 CA 9C 01           1890       JZ   SAVE2  ;YES, START WRITING HERE.
0176                    1895 *
0176                    1900 * ADVANCE TAPE TO REQUIRED FILE POSITION - 1
0176                    1905 *
0176 0E 55              1910       MVI  C,FMARK
0178 CD 5B 03           1915 ADVNC CALL GETFM  ;GET NEXT FILE MARK
017B CA 2C 02           1920       JZ   FPERR  ;JUMP IF END OF TAPE
017E 05                 1925       DCR  B      ;DECR. FILE POSITION
017F C2 78 01           1930       JNZ  ADVNC  ;LOOP UNTIL ZERO
0182
```

```
                        1935 *
0182                    1940 * READ THROUGH THIS FILE
0182                    1945 *
0182 CD 8A 03           1950       CALL GETDW ;GO PAST FILE ID/TYPE
0185 CD 8A 03           1955       CALL GETDW ;GO PAST BEGIN. ADDR
0188 CD 8A 03           1960       CALL GETDW ;GET DATA COUNT IN D-E
018B DA 13 02           1965       JC   UNCPE ;JUMP IF PARITY ERROR
018E                    1970 *
018E 21 0C 00           1975       LXI  H,TRLRL+2
0191 19                 1980       DAD  D
0192 EB                 1985       XCHG       ;DE=DE+LENGTH OF TRLR+2
0193 CD 94 03           1990 PASS  CALL GETCH ;GO PAST ALL DATA
0196 1B                 1995       DCX  D     ;DECREMENT COUNT
0197 7A                 2000       MOV  A,D
0198 B3                 2005       ORA  E     ;IS COUNT = 0 YET?
0199 C2 93 01           2010       JNZ  PASS  ;NO, CONTINUE LOOPING
019C                    2015 *
019C 11 40 01           2020 SAVE2 LXI  D,LEADL ;DE=LENGTH OF LEADER
019F                    2025 *
019F CD F0 03           2030 SAVE3 CALL SXMTC ;SEND TRANSMIT COMMAND
01A2 06 16              2035       MVI  B,SYNC ;B=SYNC WORD, DE=COUNT
01A4 CD B6 03           2040       CALL SAVBL ;WRITE BLOCK OF SYNC WORDS
01A7 CA 22 02           2045       JZ   OUTTP ;JUMP IF END OF TAPE
01AA DA 27 02           2050       JC   WPROT ;JUMP IF WRITE PROTECTED
01AD                    2055 *
01AD                    2060 * TAPE IS POSITIONED, NOW SAVE THE FILE
01AD                    2065 *
01AD 06 55              2070       MVI  B,FMARK
01AF CD CA 03           2075       CALL SAVCH ;WRITE FILE MARK
01B2 E1                 2080       POP  H     ;&FCB
01B3 3E 03              2085       MVI  A,FCBID-FCB
01B5 CD E2 02           2090       CALL EXT   ;GET FILE ID/TYPE IN D-E
01B8 E5                 2095       PUSH H
01B9 CD C3 03           2100       CALL SAVDW ;OUTPUT FILE ID/TYPE
01BC CA 22 02           2105       JZ   OUTTP ;JUMP IF END OF TAPE
01BF                    2110 *
01BF E1                 2115       POP  H     ;&FCB
01C0 3E 05              2120       MVI  A,FCBBG-FCB
01C2 CD E2 02           2125       CALL EXT   ;GET BEGIN. ADDR IN D-E
01C5 E5                 2130       PUSH H
01C6                    2135 *
01C6 01 07 00           2140       LXI  B,FCBEN-FCB
01C9 09                 2145       DAD  B     ;HL=&FCBEN
01CA 7E                 2150       MOV  A,M   ;COMPUTE HL=FCBEN-DE+1
01CB 23                 2155       INX  H
01CC 66                 2160       MOV  H,M
01CD 93                 2165       SUB  E
01CE 6F                 2170       MOV  L,A
01CF 7C                 2175       MOV  A,H
01D0 9A                 2180       SBB  D
01D1 67                 2185       MOV  H,A
01D2 23                 2190       INX  H     ;HL=DATA LENGTH
01D3
```

```
                        2195 *
01D3 0E 00              2200        MVI  C,0    ;RE-SET LONG. PARITY
01D5 CD C3 03           2205        CALL SAVDW  ;OUTPUT BEGIN. ADDR
01D8 CA 22 02           2210        JZ   OUTTP  ;JUMP IF END OF TAPE
01DB EB                 2215        XCHG        ;DE=DATA LENGTH
01DC                    2220 *                  ;HL=BEGIN. ADDR
01DC CD C3 03           2225        CALL SAVDW  ;OUTPUT DATA LENGTH
01DF CA 22 02           2230        JZ   OUTTP  ;JUMP IF END OF TAPE
01E2                    2235 *
01E2 46                 2240 SAVNX  MOV  B,M    ;B=CHAR. TO OUTPUT
01E3 CD CA 03           2245        CALL SAVCH  ;SAVE DATA FROM MEMORY
01E6 CA 22 02           2250        JZ   OUTTP  ;JUMP IF END OF TAPE
01E9 DA 27 02           2255        JC   WPROT  ;JUMP IF WRITE PROTECTED
01EC 23                 2260        INX  H      ;INCR. MEMORY POINTER
01ED 1B                 2265        DCX  D      ;DECR. COUNT
01EE 7A                 2270        MOV  A,D
01EF B3                 2275        ORA  E      ;IS COUNT = 0 YET?
01F0 C2 E2 01           2280        JNZ  SAVNX  ;NO, CONTINUE LOOPING
01F3                    2285 *
01F3 E1                 2290        POP  H      ;&FCB
01F4 3E 09              2295        MVI  A,FCBAS-FCB
01F6 CD E2 02           2300        CALL EXT    ;GET AUTOSTART IN D-E
01F9 E5                 2305        PUSH H
01FA CD C3 03           2310        CALL SAVDW  ;OUTPUT AUTOSTART ADDR
01FD CA 22 02           2315        JZ   OUTTP  ;JUMP IF END OF TAPE
0200                    2320 *
0200 41                 2325        MOV  B,C    ;B=COMPUTED LONG. PARITY
0201 11 0A 00           2330        LXI  D,TRLRL ;DE=LENGTH OF TRAILER
0204 CD B6 03           2335        CALL SAVBL  ;OUTPUT TRAILER
0207 CA 22 02           2340        JZ   OUTTP  ;JUMP IF END OF TAPE
020A CD DB 03           2345        CALL TBE    ;MAKE SURE XMTR BUFFER EMPTY
020D CD 0D 03           2350        CALL MOFF   ;TURN MOTOR OFF AND DELAY
0210 C3 37 02           2355        JMP  DONE
0213
```

```
                        2360 *
0213                    2365 * SET RETURN CODES AND STATUS FLAGS
0213                    2370 *
0213 3E 01              2375 UNCPE MVI  A,RC1 ;UNCORRECTABLE P.E.
0215 C3 2E 02           2380       JMP  ERROR
0218 3E 02              2385 NOFIL MVI  A,RC2 ;FILE NOT FOUND
021A C3 2E 02           2390       JMP  ERROR
021D 3E 03              2395 VFYER MVI  A,RC3 ;VERIFY ERROR
021F C3 2E 02           2400       JMP  ERROR
0222 3E 04              2405 OUTTP MVI  A,RC4 ;OUT OF TAPE
0224 C3 2E 02           2410       JMP  ERROR
0227 3E 05              2415 WPROT MVI  A,RC5 ;WRITE PROTECTED
0229 C3 2E 02           2420       JMP  ERROR
022C 3E 06              2425 FPERR MVI  A,RC6 ;FILE POSITION ERROR
022E                    2430 *
022E B7                 2435 ERROR ORA  A     ;RE-SET ZERO FLAG
022F 37                 2440       STC        ;SET CARRY FLAG
0230 C3 3B 02           2445       JMP  EXIT
0233                    2450 *
0233 AF                 2455 ENDTP XRA  A     ;CARRY FLAG RE-SET,
0234                    2460 *                ;ACC=0, ZERO FLAG SET
0234 C3 3B 02           2465       JMP  EXIT  ;INDICATES EOT(NOT AN ERROR)
0237                    2470 *
0237 F6 01              2475 DONE  ORI  1     ;RE-SET ZERO & CARRY FLAGS
0239 3E 00              2480       MVI  A,RC0 ;NORMAL RETURN CODE
023B                    2485 *
023B                    2490 * STANDARD EXIT FOR CAT, GET AND SAVE
023B                    2495 * ENTRY: (SP)=&FCB, RETURN CODE IN ACC.
023B                    2500 *   CARRY&ZERO FLAGS INDICATE RESULT OF OPER.
023B                    2505 * EXIT: HL=&FCB, STACK CLEANED UP
023B                    2510 *   FLAGS AND ACC. ARE PRESERVED FOR CALLER
023B                    2515 *   RETURN CODE IS INSERTED IN FCBRC
023B                    2520 *   ESF CONTROLLER IS RE-SET AND MOTOR IS OFF
023B                    2525 *
023B E1                 2530 EXIT  POP  H     ;HL=&FCBRC
023C 77                 2535       MOV  M,A   ;INSERT RETURN CODE IN FCB
023D C3 F6 03           2540       JMP  RESET ;RE-SET ESF CONTROLLER
0240
```

```
                                2545 *
0240                            2550 * MAIN PROCEDURE TO CERTIFY A TAPE.
0240                            2555 * NOTE: IF TAPE HAS A CAPACITY >= HEX FFFF
0240                            2560 * BYTES, ONLY HEX FFFE BYTES WILL BE CERTIFIED.
0240                            2565 * ENTRY: HL = LENGTH OF TAPE IN BYTES
0240                            2570 * EXIT: BC, DE PRESERVED
0240                            2575 *   ZERO & CARRY FLAGS OFF = TAPE O.K., HL=0
0240                            2580 *   ZERO FLAG SET = END OF TAPE ENCOUNTERED,
0240                            2585 *      TAPE NOT CERTIFIED, HL HAS APPROX.
0240                            2590 *      CAPACITY OF TAPE IN BYTES
0240                            2595 *   CARRY FLAG SET = NON-ZERO (ERROR) RETURN
0240                            2600 *      CODE IN ACC., HL HAS ERROR COUNT.
0240                            2605 *
0240                            2610 * FIRST, WRITE TEST WORDS ON TAPE
0240                            2615 *
0240 E5                         2620 CERT  PUSH H
0241 CD FA 02                   2625       CALL INITS ;INITIALIZE CONTROLLER
0244 CD 4E 03                   2630       CALL BOT2  ;FIND BEGIN. OF TAPE
0247 CD F0 03                   2635       CALL SXMTC ;SEND TRANSMIT COMMAND
024A                            2640 *
024A 11 40 01                   2645       LXI  D,LEADL ;LENGTH OF LEADER
024D 06 16                      2650       MVI  B,SYNC ;SYNC WORD
024F CD B6 03                   2655       CALL SAVBL ;WRITE BLOCK OF SYNC WORDS
0252 DA D7 02                   2660       JC   WFAIL ;JUMP IF WRITE FAILED
0255                            2665 *
0255 06 CC                      2670       MVI  B,CMARK
0257 CD CA 03                   2675       CALL SAVCH ;WRITE FILE MARK
025A 11 FF FF                   2680       LXI  D,0FFFFH
025D CD C3 03                   2685       CALL SAVDW ;WRITE DUMMY FILE ID
0260                            2690 *
0260 06 94                      2695       MVI  B,CCHAR ;B=TEST CHARACTER
0262 0E 00                      2700       MVI  C,0   ;C=LONG. PARITY VALUE
0264 D1                         2705       POP  D     ;DE=COUNT
0265 D5                         2710       PUSH D
0266 CD B6 03                   2715       CALL SAVBL ;FILL TAPE WITH TEST CHARS.
0269 21 4A 01                   2720       LXI  H,TRLRL+IBGPL ;HL=LENGTH OF TRAILER
026C CA C6 02                   2725       JZ   CEOT  ;JUMP IF END OF TAPE
026F                            2730 *
026F EB                         2735       XCHG       ;DE=LENGTH OF TRAILER
0270 41                         2740       MOV  B,C   ;B=COMPUTED LONG. PARITY
0271 CD B6 03                   2745       CALL SAVBL ;WRITE TRAILER
0274 21 00 00                   2750       LXI  H,0
0277 CA C6 02                   2755       JZ   CEOT  ;JUMP IF END OF TAPE
027A                            2760 *
027A D1                         2765       POP  D
027B D5                         2770       PUSH D
027C 13                         2775       INX  D
027D 7B                         2780       MOV  A,E   ;AT FIRST PASS, DE=0FFFFH
027E B2                         2785       ORA  D     ;IS THIS THE FIRST PASS?
027F C2 8A 02                   2790       JNZ  CERT1 ;NO
0282 D1                         2795       POP  D
0283 1B                         2800       DCX  D     ;YES, SET CAPACITY TO 0FFFEH
0284 D5                         2805       PUSH D
0285 13                         2810       INX  D
0286 13                         2815       INX  D     ;DE=0
0287 C3 C6 02                   2820       JMP  CEOT
028A
```

```
                        2825 *
028A                    2830 * READ BACK AND VERIFY TEST WORDS
028A                    2835 *
028A CD DB 03           2840 CERT1 CALL TBE   ;MAKE SURE XMTR BUFFER EMPTY
028D 0E CC              2845       MVI  C,CMARK
028F 11 FF FF           2850       LXI  D,0FFFFH
0292 CD 1F 03           2855       CALL FIND  ;FIND DUMMY FILE ID
0295 3E 02              2860       MVI  A,RC2
0297 DA D9 02           2865       JC   CERR1 ;JUMP IF NOT FOUND
029A                    2870 *
029A D1                 2875       POP  D     ;DE=DATA COUNT
029B D5                 2880       PUSH D
029C AF                 2885       XRA  A     ;ACC.=0
029D 47                 2890       MOV  B,A   ;B=PARITY ERROR COUNT
029E 4F                 2895       MOV  C,A   ;C=LONG. PARITY VALUE
029F 67                 2900       MOV  H,A
02A0 6F                 2905       MOV  L,A   ;HL=VERIFY ERROR COUNT
02A1 CD 94 03           2910 CERT2 CALL GETCH ;GET DATA CHAR.
02A4 FE 94              2915       CPI  CCHAR ;IS IT THE TEST CHARACTER?
02A6 CA AA 02           2920       JZ   CERT3 ;YES
02A9 23                 2925       INX  H     ;NO, INCR. ERROR COUNT
02AA 1B                 2930 CERT3 DCX  D     ;DECR. DATA COUNT
02AB 7A                 2935       MOV  A,D
02AC B3                 2940       ORA  E     ;IS COUNT = 0 YET?
02AD C2 A1 02           2945       JNZ  CERT2 ;NO, CONTINUE LOOPING
02B0                    2950 *
02B0 51                 2955       MOV  D,C   ;D=COMPUTED LONG. PARITY
02B1 CD 94 03           2960       CALL GETCH ;GET LONG. PARITY WORD
02B4 BA                 2965       CMP  D     ;SAME AS COMPUTED?
02B5 C2 D2 02           2970       JNZ  PAERR ;NO
02B8 78                 2975       MOV  A,B   ;CHECK FOR ERRORS
02B9 B1                 2980       ORA  C     ;BC, HL SHOULD = 0
02BA B4                 2985       ORA  H
02BB B5                 2990       ORA  L
02BC C2 D2 02           2995       JNZ  PAERR ;JUMP IF ANY ERRORS
02BF 3C                 3000       INR  A     ;RE-SET ZERO & CARRY FLAGS
02C0 3E 00              3005       MVI  A,RC0 ;NORMAL RETURN CODE
02C2 D1                 3010       POP  D     ;THROW AWAY ORIGINAL COUNT
02C3 C3 F6 03           3015       JMP  RESET
02C6
```

```
                        3020 *
                        3025 * COMPUTE APPROX. CAPACITY (IN BYTES) OF TAPE
02C6                    3030 *
02C6 19                 3035 CEOT  DAD  D      ;HL=DEFICIT
02C7 D1                 3040       POP  D      ;DE=ESTIMATED CAPACITY
02C8                    3045 *
02C8 7B                 3050       MOV  A,E    ;COMPUTE HL=DE-HL
02C9 95                 3055       SUB  L
02CA 6F                 3060       MOV  L,A
02CB 7A                 3065       MOV  A,D
02CC 9C                 3070       SBB  H
02CD 67                 3075       MOV  H,A    ;HL=APPROX. CAPACITY
02CE AF                 3080       XRA  A      ;ACC.=0,Z SET, C RE-SET
02CF C3 F6 03           3085       JMP  RESET
02D2                    3090 *
02D2 3E 01              3095 PAERR MVI  A,RC1  ;UNCORRECTABLE P.E.
02D4 C3 DC 02           3100       JMP  CERR2
02D7 3E 05              3105 WFAIL MVI  A,RC5  ;WRITE PROTECTED
02D9 21 00 00           3110 CERR1 LXI  H,0
02DC B7                 3115 CERR2 ORA  A      ;RE-SET ZERO FLAG
02DD 37                 3120       STC         ;SET CARRY FLAG
02DE D1                 3125       POP  D      ;THROW AWAY ORIGINAL COUNT
02DF C3 F6 03           3130       JMP  RESET  ;RE-SET ESF CONTROLLER
02E2                    3135 *
02E2                    3140 * EXTRACT 2-BYTE VALUE FROM FCB
02E2                    3145 * ENTRY: ACC = OFFSET; HL = &FCB
02E2                    3150 * EXIT: DE = VALUE; BC,HL PRESERVED
02E2                    3155 *
02E2 E5                 3160 EXT   PUSH H
02E3 85                 3165       ADD  L      ;HL=HL+ACC
02E4 6F                 3170       MOV  L,A
02E5 3E 00              3175       MVI  A,0
02E7 8C                 3180       ADC  H
02E8 67                 3185       MOV  H,A
02E9 5E                 3190       MOV  E,M    ;DE=(HL)
02EA 23                 3195       INX  H
02EB 56                 3200       MOV  D,M
02EC E1                 3205       POP  H
02ED C9                 3210       RET
02EE                    3215 *
02EE                    3220 * INSERT 2-BYTE VALUE INTO FCB.
02EE                    3225 * ENTRY: ACC = OFFSET; DE = VALUE; HL = &FCB
02EE                    3230 * EXIT: BC,DE,HL PRESERVED
02EE                    3235 *
02EE E5                 3240 INS   PUSH H
02EF 85                 3245       ADD  L      ;HL=HL+ACC
02F0 6F                 3250       MOV  L,A
02F1 3E 00              3255       MVI  A,0
02F3 8C                 3260       ADC  H
02F4 67                 3265       MOV  H,A
02F5 73                 3270       MOV  M,E    ;(HL)=DE
02F6 23                 3275       INX  H
02F7 72                 3280       MOV  M,D
02F8 E1                 3285       POP  H
02F9 C9                 3290       RET
02FA
```

```
                        3295 *
02FA                    3300 * INITIALIZE SYSTEM AND START MOTOR.
02FA                    3305 * EXIT: HL = &CONTROL PORT; BC = 0
02FA                    3310 *
02FA 3A FB 0F           3315 INITS LDA  SWTPT ;FORCE SINGLE DENSITY
02FD 21 FF 0F           3320       LXI  H,CTLPT ;HL=&CONTROL PORT
0300 AF                 3325       XRA  A     ;DUMMY WORD
0301 77                 3330       MOV  M,A   ;SEND IT 3 TIMES
0302 77                 3335       MOV  M,A
0303 77                 3340       MOV  M,A
0304 36 40              3345       MVI  M,RSTC ;SEND RESET COMMAND
0306 36 BC              3350       MVI  M,MODC ;SEND MODE COMMAND
0308 36 16              3355       MVI  M,SYNC ;SEND SYNC WORD
030A 36 12              3360       MVI  M,MTRC ;SEND MOTOR-ON COMMAND
030C C9                 3365       RET
030D                    3370 *
030D                    3375 * MOTOR-OFF AND DELAY ROUTINE
030D                    3380 *
030D 3E 31              3385 MOFF  MVI  A,MOFC ;TURN MOTOR OFF
030F 32 FF 0F           3390       STA  CTLPT  ;(BUT KEEP SENDING)
0312                    3395 *
0312 01 00 20           3400       LXI  B,8192 ;FOR 200 MSEC. DELAY
0315 C5                 3405 DELAY PUSH B     ;THIS DELAY LOOP OPERATES
0316 C1                 3410       POP  B     ;IN INCREMENTS OF 25 USEC.
0317 0B                 3415       DCX  B     ;(AT 2 MHZ CLOCK RATE)
0318 78                 3420       MOV  A,B   ;THE NO. OF INCREMENTS IS IN
0319 78                 3425       MOV  A,B   ;B-C AT START OF LOOP.
031A B1                 3430       ORA  C
031B C2 15 03           3435       JNZ  DELAY
031E C9                 3440       RET
031F
```

```
                        3445 *
031F                    3450 * FIND THE REQUIRED FILE ON THE TAPE.
031F                    3455 * ENTRY: D = FILE TYPE, E = FILE ID TO FIND
031F                    3460 *         C = FILE MARK
031F                    3465 * EXIT: BHL DESTROYED, CDE PRESERVED
031F                    3470 *   CARRY FLAG OFF = TAPE IS CORRECTLY POS-
031F                    3475 *      ITIONED TO READ THE BEGINNING ADDRESS
031F                    3480 *      FROM THE REQUIRED FILE.
031F                    3485 *   CARRY FLAG SET = FILE NOT FOUND
031F                    3490 *
031F 06 00              3495 FIND  MVI  B,0    ;RE-SET END-OF-TAPE SWITCH
0321 CD 5B 03           3500 FIND1 CALL GETFM  ;GET NEXT FILE MARK
0324 C2 31 03           3505       JNZ  FIND3  ;JUMP IF NOT EOT
0327 AF                 3510 FIND2 XRA  A      ;ACC=0
0328 B8                 3515       CMP  B      ;HAS EOT BEEN PASSED?
0329 D8                 3520       RC          ;YES, RETURN CARRY SET.
032A 04                 3525       INR  B      ;NO, SET EOT SWITCH.
032B CD 4B 03           3530       CALL BOT1   ;FIND BEGINNING OF TAPE
032E C3 21 03           3535       JMP  FIND1
0331 CD 75 03           3540 FIND3 CALL GETWD  ;GET FILE ID
0334 CA 27 03           3545       JZ   FIND2  ;JUMP IF EOT
0337 6F                 3550       MOV  L,A    ;L=FILE ID
0338 CD 75 03           3555       CALL GETWD  ;GET FILE TYPE
033B CA 27 03           3560       JZ   FIND2  ;JUMP IF EOT
033E 67                 3565       MOV  H,A    ;H=FILE TYPE
033F BA                 3570       CMP  D      ;CORRECT FILE TYPE?
0340 C2 21 03           3575       JNZ  FIND1  ;NO, KEEP LOOKING
0343 7B                 3580       MOV  A,E
0344 BD                 3585       CMP  L      ;CORRECT FILE ID?
0345 C8                 3590       RZ          ;YES
0346 B7                 3595       ORA  A      ;REQ'D FILE ID = 0?
0347 C2 21 03           3600       JNZ  FIND1  ;NO, KEEP LOOKING
034A C9                 3605       RET         ;YES, ACCEPT ANY FILE ID.
034B
```

```
                        3610 *
034B                    3615 * FIND BEGINNING OF TAPE
034B                    3620 * ENTRY: HL = &CONTROL PORT, NOT CHANGED
034B                    3625 *
034B 21 FF 0F           3630 BOT1  LXI  H,CTLPT ;HL=&CTLPT
034E 7E                 3635 BOT2  MOV  A,M     ;GET STATUS
034F E6 80              3640       ANI  EOTM    ;END OF TAPE?
0351 C2 4E 03           3645       JNZ  BOT2    ;NO, LOOP AGAIN
0354 7E                 3650 BOT3  MOV  A,M     ;GET STATUS
0355 E6 80              3655       ANI  EOTM    ;END-OF-TAPE MARKER?
0357 CA 54 03           3660       JZ   BOT3    ;YES, LOOP AGAIN
035A C9                 3665       RET
035B                    3670 *
035B                    3675 * GET NEXT BEGINNING-OF-FILE MARK
035B                    3680 * ENTRY: REG. C HAS FILE MARK CHARACTER
035B                    3685 * EXIT: ZERO FLAG OFF = FILE MARK FOUND
035B                    3690 *       ZERO FLAG SET = END OF TAPE
035B                    3695 *
035B 3E 96              3700 GETFM MVI  A,HUNC
035D 32 FF 0F           3705       STA  CTLPT   ;SEND HUNT COMMAND
0360 3A FE 0F           3710       LDA  DATPT   ;RE-SET STATUS
0363 CD 75 03           3715 GETF2 CALL GETWD   ;GET DATA WORD
0366 C8                 3720       RZ           ;RETURN IF EOT
0367 B9                 3725       CMP  C       ;FILE MARK?
0368 CA 73 03           3730       JZ   GETFX   ;YES
036B FE 16              3735       CPI  SYNC    ;SYNC WORD?
036D CA 63 03           3740       JZ   GETF2   ;YES
0370 C3 5B 03           3745       JMP  GETFM   ;NO
0373 B7                 3750 GETFX ORA  A       ;RE-SET ZERO FLAG
0374 C9                 3755       RET
0375                    3760 *
0375                    3765 * GET DATA WORD FROM TAPE TO ACC.
0375                    3770 * DOES NOT CHECK FOR PARITY ERRORS
0375                    3775 * EXIT: ZERO FLAG OFF = DATA IN ACC.
0375                    3780 *       ZERO FLAG SET = END OF TAPE
0375                    3785 *
0375 C5                 3790 GETWD PUSH B
0376 3A FF 0F           3795 GETW2 LDA  CTLPT   ;GET STATUS
0379 47                 3800       MOV  B,A     ;SAVE IT
037A E6 80              3805       ANI  EOTM    ;END OF TAPE?
037C CA 88 03           3810       JZ   GETWX   ;YES, RETURN Z SET
037F 78                 3815       MOV  A,B     ;GET STATUS
0380 E6 02              3820       ANI  RDYM    ;DATA READY?
0382 CA 76 03           3825       JZ   GETW2   ;NO, LOOP AGAIN
0385 3A FE 0F           3830       LDA  DATPT   ;GET DATA IN ACC.
0388 C1                 3835 GETWX POP  B
0389 C9                 3840       RET
038A
```

```
                        3845 *
038A                    3850 * GET DOUBLE WORD FROM TAPE TO D-E REGS.
038A                    3855 * CALLS GETCH, CHECKS FOR PARITY ERRORS.
038A                    3860 *
038A CD 94 03           3865 GETDW CALL GETCH ;GET LOW-ORDER BYTE
038D 5F                 3870       MOV  E,A   ;IN REG. E
038E D8                 3875       RC         ;RETURN ON ERROR
038F CD 94 03           3880       CALL GETCH ;GET HIGH-ORDER BYTE
0392 57                 3885       MOV  D,A   ;IN REG. D
0393 C9                 3890       RET
0394                    3895 *
0394                    3900 * GET DATA CHARACTER FROM TAPE TO ACC.
0394                    3905 * THIS ROUTINE CHECKS FOR PARITY ERRORS
0394                    3910 * ENTRY: REG. B = PARITY ERROR COUNT
0394                    3915 *         REG. C = LONGITUDINAL PARITY VALUE
0394                    3920 * EXIT: B AND C ARE UPDATED; DE, HL PRESERVED
0394                    3925 *       CARRY FLAG OFF = DATA IN ACC.
0394                    3930 *       CARRY FLAG SET = PARITY ERROR
0394                    3935 *
0394 E5                 3940 GETCH PUSH H
0395 21 FF 0F           3945       LXI  H,CTLPT
0398 7E                 3950 GETC2 MOV  A,M   ;GET STATUS
0399 E6 02              3955       ANI  RDYM  ;DATA READY?
039B CA 98 03           3960       JZ   GETC2 ;NO, TRY AGAIN
039E 7E                 3965       MOV  A,M   ;GET STATUS
039F E6 08              3970       ANI  PAEM  ;PARITY ERROR?
03A1 CA AE 03           3975       JZ   GETC3  ;NO, CARRY IS RE-SET
03A4 36 16              3980       MVI  M,RCVC ;YES, RE-SET P/E FLAG
03A6 78                 3985       MOV  A,B
03A7 FE FF              3990       CPI  255   ;IS P.E. COUNT >= 255?
03A9 D2 AD 03           3995       JNC  PE1   ;YES
03AC 04                 4000       INR  B     ;NO, INCR. P.E. COUNT
03AD 37                 4005 PE1   STC        ;SET CARRY FLAG
03AE 2B                 4010 GETC3 DCX  H     ;HL=&DATA PORT
03AF 7E                 4015       MOV  A,M   ;GET DATA IN ACC.
03B0 F5                 4020       PUSH PSW   ;SAVE FLAGS & DATA
03B1 A9                 4025       XRA  C
03B2 4F                 4030       MOV  C,A   ;UPDATE LONG. PARITY
03B3 F1                 4035       POP  PSW   ;RESTORE FLAGS & DATA
03B4 E1                 4040       POP  H
03B5 C9                 4045       RET
03B6
```

```
                        4050 *
03B6                    4055 * AT EXIT FROM ALL TAPE OUTPUT ROUTINES:
03B6                    4060 * ZERO AND CARRY FLAGS OFF = SUCCESSFUL OPER.
03B6                    4065 * ZERO FLAG SET = END OF TAPE
03B6                    4070 * CARRY FLAG SET = WRITE PROTECTED
03B6                    4075 *
03B6                    4080 * SAVE A BLOCK OF LEADER, TRAILER OR SYNC WDS
03B6                    4085 * ENTRY: B = DATA TO OUTPUT
03B6                    4090 *         C = LONGITUDINAL PARITY VALUE
03B6                    4095 *         DE = COUNT (ASSUMED > 0)
03B6                    4100 *         HL = NOT USED
03B6                    4105 * EXIT: B PRESERVED, C UPDATED, D-E ZEROED
03B6                    4110 *
03B6 CD CA 03           4115 SAVBL CALL SAVCH ;SAVE ONE CHAR.
03B9 C8                 4120       RZ         ;RETURN IF EOT
03BA D8                 4125       RC         ;RETURN IF WRITE FAILED
03BB 1B                 4130       DCX  D     ;DECR. COUNT
03BC 7A                 4135       MOV  A,D
03BD B3                 4140       ORA  E     ;IS COUNT = 0 YET?
03BE C2 B6 03           4145       JNZ  SAVBL ;NO, CONTINUE LOOPING
03C1 3C                 4150       INR  A     ;RE-SET ZERO AND CARRY
03C2 C9                 4155       RET
03C3                    4160 *
03C3                    4165 * SAVE A TWO-BYTE VALUE
03C3                    4170 * ENTRY: BC = SAME AS ABOVE, HL = NOT USED
03C3                    4175 *   DE = VALUE TO OUTPUT (E FIRST), PRESERVED
03C3                    4180 *
03C3 43                 4185 SAVDW MOV  B,E
03C4 CD CA 03           4190       CALL SAVCH ;SAVE LOW-ORDER BYTE
03C7 C8                 4195       RZ         ;RETURN IF EOT
03C8 D8                 4200       RC         ;RETURN IF WRITE FAILED
03C9 42                 4205       MOV  B,D
03CA                    4210 *                ;SAVE HIGH-ORDER BYTE
03CA                    4215 *
03CA                    4220 * SAVE A DATA CHARACTER ON TAPE.
03CA                    4225 * ENTRY: B = DATA TO OUTPUT,PRESERVED AT EXIT
03CA                    4230 *         C = LONGITUDINAL PARITY,GETS UPDATED
03CA                    4235 *         DE, HL NOT USED
03CA                    4240 *
03CA C5                 4245 SAVCH PUSH B
03CB CD DB 03           4250       CALL TBE   ;WAIT FOR XMTR BUFFER EMPTY
03CE C1                 4255       POP  B
03CF C8                 4260       RZ         ;RETURN IF EOT
03D0 D8                 4265       RC         ;RETURN IF WRITE PROTECTED
03D1 78                 4270       MOV  A,B
03D2 A9                 4275       XRA  C
03D3 4F                 4280       MOV  C,A   ;UPDATE LONG. PARITY VALUE
03D4 78                 4285       MOV  A,B   ;DATA TO OUTPUT
03D5 32 FE 0F           4290       STA  DATPT ;OUTPUT TO DATA PORT
03D8 F6 FF              4295       ORI  0FFH  ;RE-SET ZERO & CARRY FLAGS
03DA C9                 4300       RET
03DB
```

```
                        4305 *
03DB                    4310 * WAIT FOR TRANSMITTER BUFFER EMPTY
03DB                    4315 * EXIT: REG. C DESTROYED!
03DB                    4320 *
03DB 0E 81              4325 TBE   MVI  C,129 ;INITIALIZE TIMER VALUE
03DD 3A FF 0F           4330 TBEL  LDA  CTLPT ;GET STATUS
03E0 E6 80              4335       ANI  EOTM  ;END OF TAPE?
03E2 C8                 4340       RZ         ;YES, RETURN Z FLAG SET.
03E3 3A FF 0F           4345       LDA  CTLPT ;GET STATUS
03E6 E6 01              4350       ANI  TBEM  ;XMTR BUFFER EMPTY?
03E8 C0                 4355       RNZ        ;YES, RETURN Z&C RE-SET
03E9 0D                 4360       DCR  C     ;DECREMENT TIMER
03EA C2 DD 03           4365       JNZ  TBEL  ;CONTINUE LOOPING IF NOT 0
03ED 0C                 4370       INR  C     ;RE-SET ZERO FLAG
03EE 37                 4375       STC        ;SET CARRY FLAG
03EF C9                 4380       RET
03F0                    4385 *
03F0                    4390 * SEND TRANSMIT COMMAND
03F0                    4395 *
03F0 3E 33              4400 SXMTC MVI  A,XMTC
03F2 32 FF 0F           4405       STA  CTLPT
03F5 C9                 4410       RET
03F6                    4415 *
03F6                    4420 * RE-SET CONTROLLER, TURN MOTOR OFF
03F6                    4425 *
03F6 F5                 4430 RESET PUSH PSW
03F7 3E 40              4435       MVI  A,RSTC
03F9 32 FF 0F           4440       STA  CTLPT
03FC F1                 4445       POP  PSW
03FD C9                 4450       RET
03FE                    4455 *
03FE FF FF              4460       DW   OFFFFH ;NOT USED
0400                    4465 *
0400                    4470 * DEFINITION OF FILE CONTROL BLOCK (FCB)
0400                    4475 * THIS SPACE IS NOT ACTUALLY USED BY THE
0400                    4480 * ESF UTILITY.  INSTEAD, THE CALLER SUP-
0400                    4485 * PLIES ITS OWN FCB WHICH CAN BE ANYWHERE
0400                    4490 * IN MEMORY.
0400                    4495 *
0400                    4500 FCB   EQU  $
0400                    4505 FCBRC DS   1 ;RETURN CODE
0401                    4510 FCBOP DS   1 ;OPTION (GET, CAT)
0402                    4515 FCBFP DS   1 ;FILE POSITION (SAVE, CAT)
0403                    4520 FCBID DS   1 ;FILE ID
0404                    4525 FCBTY DS   1 ;FILE TYPE
0405                    4530 FCBBG DS   2 ;BEGIN. ADDRESS L-H
0407                    4535 FCBEN DS   2 ;ENDING ADDRESS L-H
0409                    4540 FCBAS DS   2 ;AUTOSTART ADDRESS L-H
040B                    4545 FCBER DS   2 ;DATA ERROR ADDRESS L-H
040D                    4550 FCBLN EQU  $ ;FCBLN-FCB = LENGTH OF FCB
040D
```

```
                        4555 *

ADVNC   0178    1930
BOT1    034B    0935 1880 3530
BOT2    034E    0460 2630 3645
BOT3    0354    3660
CAT     006B    0410
CCHAR   0094    2695 2915
CEOT    02C6    2725 2755 2820
CERR1   02D9    2865
CERR2   02DC    3100
CERT    0240    0425
CERT1   028A    2790
CERT2   02A1    2945
CERT3   02AA    2920
CMARK   00CC    2670 2845
CTLPT   0FFF    3320 3390 3630 3705 3795 3945 4330 4345 4405 4440
DATPT   0FFE    3710 3830 4290
DELAY   0315    3435
DONE    0237    1580 1670 1680 1720 2355
ECAT    0008
ECER    0011
EGET    000B
EIPL    0000
EMON    0004
ENDTP   0233    0955
EOTM    0080    3640 3655 3805 4335
ERROR   022E    2380 2390 2400 2410 2420
ESAV    000E
ESFCT   0000
EWTP    0014
EXIT    023B    2445 2465
EXT     02E2    0705 1060 1130 1695 2090 2125 2300
FCB     0400    0530 0700 0990 1055 1125 1225 1310 1480 1515 1690 1740
                2085 2120 2140 2295
FCBAS   0409    0700 1515 2295
FCBBG   0405    1125 1225 2120
FCBEN   0407    1480 2140
FCBER   040B    1310 1690 1740
FCBFP   0402
FCBID   0403    0990 1055 2085
FCBLN   040D    0530
FCBOP   0401
FCBRC   0400
FCBTY   0404
FILER   0153    1405
FIND    031F    1080 2855
FIND1   0321    3535 3575 3600
FIND2   0327    3545 3560
FIND3   0331    3505
FMARK   0055    0945 1075 1910 2070
FPERR   022C    1920
```

GET	0098	0415 0680
GET2	00AF	1005
GET3	00C9	1210
GET4	00DE	1305
GET5	00F5	1365 1385
GETC2	0398	3960
GETC3	03AE	3975
GETCH	0394	1275 1565 1600 1990 2910 2960 3865 3880
GETDW	038A	1170 1250 1500 1950 1955 1960
GETF2	0363	3740
GETFM	035B	0950 1915 3500 3745
GETFX	0373	3730
GETNX	00CF	1455
GETW2	0376	3825
GETWD	0375	0965 0975 3540 3555 3715
GETWX	0388	3810
HALT	0068	0785
HUNC	0096	3700
IBGPL	0140	1850 2720
INITS	02FA	0455 0885 1070 1820 2625
INS	02EE	0995 1230 1315 1485 1520 1745
IPL	001D	0385
IPLER	0065	0580 0685
LEADL	0140	2020 2645
MODC	00BC	3350
MOFC	0031	3385
MOFF	030D	2350
MTRC	0012	3360
NOAS	0062	0735
NOFIL	0218	1085
ONEPE	0129	1550
OUTTP	0222	2045 2105 2210 2230 2250 2315 2340
PAEM	0008	3970
PAERR	02D2	2970 2995
PASS	0193	2010
PE1	03AD	3995
PSW	0006	4020 4035 4430 4445
RAMCH	006D	0560
RC0	0000	2480 3005
RC1	0001	2375 3095
RC2	0002	2385 2860
RC3	0003	2395
RC4	0004	2405
RC5	0005	2415 3105
RC6	0006	2425
RC7	0007	0765
RC8	0008	0575

```
RCVC    0016    3980
RDYM    0002    3820 3955
RESET   03F6    0465 2540 3015 3085 3130
ROLL    0132    1630
RSTC    0040    3345 4435
SAVBL   03B6    2040 2335 2655 2715 2745 4145
SAVCH   03CA    2075 2245 2675 4115 4190
SAVDW   03C3    2100 2205 2225 2310 2685
SAVE    015E    0420
SAVE2   019C    1890
SAVE3   019F    1865
SAVNX   01E2    2280
SP      0006
START   0000    0400
SWTPT   0FFB    0485 3315
SXMTC   03F0    2030 2635
SYNC    0016    2035 2650 3355 3735
TBE     03DB    2345 2840 4250
TBEL    03DD    4365
TBEM    0001    4350
TRLRL   000A    1975 2330 2720
UNCPE   0213    1175 1255 1505 1555 1575 1610 1640 1965
VERFY   00F0    1355
VFYER   021D    1775
WFAIL   02D7    2660
WPROT   0227    2050 2255
XMTC    0033    4400

FCHK

A     2A00  9FF5
EXEC C004
```